EFFECTIVE CHANNEL ESTIMATION AND EFFICIENT SYMBOL DETECTION FOR MULTI-INPUT MULTI-OUTPUT UNDERWATER ACOUSTIC COMMUNICATIONS

By

JUN LING

A DISSERTATION PRESENTED TO THE GRADUATE SCHOOL
OF THE UNIVERSITY OF FLORIDA IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

UNIVERSITY OF FLORIDA

2011

ACKNOWLEDGMENTS

Foremost, I would like to recognize my advisor, Dr. Jian Li of the Electrical Engineering Department, for her excellent guidance, whole-hearted dedication, and continuous encouragement throughout my Ph.D. training. I feel extremely lucky to be involved in the underwater acoustic communication project and have her as a source of wisdom, skills, and encourages. Without her help, this dissertation would not have been possible. I furthermore acknowledge Dr. Magnus Nordenvaad of Lulea University of technology, whose constant assistance and constructive comments on my work have certainly made me into a better thinker and engineer. I also would like to make special reference to Dr. James Preisig of Woods Hole Oceanographic Institution, for his help on experimental data collection. I recognize my committee members at the University of Florida: Dr. Yijun Sun, Dr. Jenshan Lin, and Dr. Louis N. Cattafesta III. Their time and efforts towards my dissertation and defense have been greatly appreciated. Last, but not least, I thank my family and my friends in the Spectral Analysis Laboratory. Their encouragement has inspired me at every level of my graduate studies.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Abstract of Dissertation Presented to the Graduate School
of the University of Florida in Partial Fulfillment of the
Requirements for the Degree of Doctor of Philosophy

EFFECTIVE CHANNEL ESTIMATION AND EFFICIENT SYMBOL DETECTION FOR MULTI-INPUT MULTI-OUTPUT UNDERWATER ACOUSTIC COMMUNICATIONS

By

Jun Ling

August 2011

Chair: Jian Li
Major: Electrical and Computer Engineering

Achieving reliable underwater acoustic communications (UAC) has long been recognized as a challenging problem owing to the scarce bandwidth available and the reverberant spread in both time and frequency domains. To pursue high data rates, we consider a multi-input multi-output (MIMO) UAC system, and our focus is placed on two main issues regarding a MIMO UAC system: 1) channel estimation, which involves the design of the training sequences and the development of a reliable channel estimation algorithm, and 2) symbol detection, which requires interference cancelation schemes due to simultaneous transmission from multiple transducers.

To enhance channel estimation performance, we present a cyclic approach for designing training sequences with good auto- and cross-correlation properties, and a channel estimation algorithm called the iterative adaptive approach (IAA). Sparse channel estimates can be obtained by combining IAA with the Bayesian information criterion (BIC). Moreover, we present sparse learning via iterative minimization (SLIM) and demonstrate that SLIM gives similar performance to IAA but at a much lower computational cost. Furthermore, an extension of the SLIM algorithm is introduced to estimate the sparse and frequency modulated acoustic channels. The extended algorithm is referred to as generalization of SLIM (GoSLIM). Regarding symbol detection, a linear minimum mean-squared error based detection scheme, called RELAX-BLAST, which is a combination of vertical Bell Labs layered space-time

(V-BLAST) algorithm and the cyclic principle of the RELAX algorithm, is presented and it is shown that RELAX-BLAST outperforms V-BLAST. We show that RELAX-BLAST can be implemented efficiently by making use of the conjugate gradient method and diagonalization properties of circulant matrices. This fast implementation approach requires only simple fast Fourier transform operations and facilitates parallel implementations. The effectiveness of the proposed MIMO schemes is verified by both computer simulations and experimental results obtained by analyzing the measurements acquired in multiple in-water experiments.

CHAPTER 1
INTRODUCTION

Earth is a water planet: approximately 71% of its surface is covered by water. Despite countless attempts to explore this vast mysterious underwater frontier throughout human history, the majority of the ocean body still remains unexplored. The advances over the last several decades in hardware and communication techniques have led to a recent excitement in underwater activities, including environmental monitoring, commercial or research exploration, and harbor protection. For these tasks, preferable systems involve the deployment of underwater sensors or the employment of unmanned underwater vehicles (UUVs). (Figure 1-1 shows a UUV.) As a consequence, the establishment of underwater acoustic communications (UAC) is critical to ensure reliable data exchange among these free underwater nodes, which makes it possible to further coordinate them. For example, had reliable UAC techniques been in use, UUVs would have been employed to control the oil spill in the Gulf of Mexico in April 2010 (Figure 1-2).

Since water is not a medium suitable for propagating electromagnetic waves, UAC has to rely on acoustic waves to transmit signals. In contrast to radio communications, which have already made significant impacts upon everyday life, the development of UAC is still in the research stage mainly due to the unique challenges imposed by the underwater environment. Four major challenges will be elaborated in Section 1.1, along with an explanation of why it is difficult to achieve reliable UAC. Section 1.2 briefly reviews the existing UAC schemes. Section 1.3 presents the outline of this dissertation. Section 1.4 lists the notations used throughout this dissertation.

1.1 Challenges of Underwater Acoustic Communications (UAC)

To help understand the unique challenges posed by the underwater environment, Table 1-1 contrasts key system parameters employed by the wireless local area network (WLAN) standard for radio communications [87] and several recently conducted

in-water UAC experiments. The experimental specifics of the four in-water UAC experiments mentioned, namely RACE08, SPACE08, WHOI09, and ACOMM10, will be further elaborated in the subsequent chapters, along with a detailed summary of the experimental results.

One of the defining characteristics of an acoustic channel is that the absorption of the underwater medium increases as the signal frequency increases [76]. The relationship between absorption coefficient and frequency is shown in Figure 1-3. One observes that due to severe absorption (or strong signal attenuation), the output power of a signal frequency modulated at 180 k Hz will reduce by almost 50 dB over 1 km propagation. As a consequence, the power of the received signal in response to the 180 k Hz transmitted signal could be too weak to ensure reliable UAC at any reasonable distance. To mitigate the power absorption, practical UAC systems adopt relatively low carrier frequency and limited signal bandwidth, over which the frequency response is relatively flat [76]. This explains why the bandwidth and carrier frequency in UAC are so small compared to those employed by WLAN (Table 1-1). Roughly speaking, the bandwidth of the transmitted signal equals the symbol rate. The limited bandwidth available, therefore, imposes an upper bound on the attainable symbol rate in conventional single-input systems. To overcome this restriction, through simultaneous transmission using multiple transmitters, multi-input multi-output (MIMO) systems offer increased data rates compared to their single-input counterparts [84]. A detailed study of MIMO UAC systems, the design of the simultaneously transmitted sequence set, and the development of effective channel estimation algorithms and efficient symbol detection schemes, forms the focus of this dissertation.

In a shallow water environment, due to the reflections from both the surface and bottom, the transmitted signal can reach the receiving hydrophone via different propagation paths at different delays [55]. Figure 1-4A illustrates an acoustic channel characterized by three multipaths, namely a direct path (or principal arrival), a surface

Table 1-1. The comparison of key system parameters employed in UAC and WLAN.

	RACE08	SPACE08	WHOI09	ACOMM10	WLAN
Average delay spread	on the order of 10 ms				$\sim$ 500 ns
Propagation speed	1500 m/s				3×10^8 m/s
Bandwidth	3.9 kHz	10 kHz	8 kHz	4 kHz	20 MHz
Carrier frequency	12 kHz	13 kHz	30 kHz	20 kHz	5.2 GHz
Coherence time (in symbols)	$\sim 10^3$				$\sim 10^5$

reflection, and a bottom reflection. The empirical underwater channel, of course, is much more complex than this simple model by allowing for more reflection combinations, including surface-bottom reflection, bottom-surface reflection, and so on [55]. Multipath propagation, coupled with the relatively low velocity of acoustic waves compared to electromagnetic waves, leads to a large spread in delay, as shown in Table 1-1. The difference in the propagation time between the earliest and latest arrivals (counting the paths with significant powers only) could span tens to hundreds of symbol periods, which translates into long channel impulse response (CIR) and severe inter-symbol interference (ISI) at the receiver side. A typical CIR estimated in an empirical underwater environment is shown in Figure 1-4B, where 80 channel taps are considered. Besides the ISI, MIMO transmission further introduces severe interference from all other transmitters, which significantly complicates the structure of the receiver and makes difficult the extraction of the desired symbols from any given transmitter.

Acoustic channels are well known as double spreading channels. That is, besides the spreading in the time domain (i.e., the long delay spread, as previously remarked), the acoustic channel also spreads in the frequency domain, notably, the Doppler effects [6, 55]. The presence of the Doppler effects, owing to the relative motions between the transmitters and receiver platforms and the dynamic underwater medium, induces a frequency-dependent phase shift to the transmitted symbols or even signal scaling (i.e., signal compressing or stretching). A Doppler-induced phase shift or signal scaling impairs the reliability of UAC, especially in the case of a phase-coherent detection scheme. A preferable tool for characterizing a double spreading channel is the scattering function, which decouples the acoustic channel into a bank of paths that experience different delays and Doppler frequencies [43]. Figure 1-5 shows two scattering functions obtained in two different sea conditions. Figure 1-5A shows that below frozen surface of the Arctic Ocean, both the direct path at 5 ms delay and the surface reflection at 10 ms delay are centered at 0 Hz. This observation suggests that the underlying acoustic

channel suffers from negligible Doppler spreading, and therefore, the channel can be reasonably modeled as an ISI channel (i.e., spreading in delay only). In contrast, in the Bahamas on a windy day, Figure 1-5B demonstrates that both the direct path and the surface reflection experience significant Doppler shifts. Most notably, the span of the Doppler frequency experienced by the surface reflections exceeds 10 Hz. In this scenario, the Doppler spreading cannot be ignored, and the underlying channel is indeed double spreading.

On top of the scarce bandwidth available and double spreading, the underwater acoustic channel is also time-varying in nature. By lining up a series of CIRs estimated at a regular period (every 38.4 ms in this example), Figure 1-6 shows the evolution of the normalized CIR over approximately a one-minute period. The CIR estimate at the 0 s reference time is shown in Figure 1-4B. One observes from Figure 1-6 that the channel taps, especially those corresponding to surface-interactive paths after 5 ms delay, experience significant variations over time. Compared to radio frequency wireless communications, the highly time-varying nature of the acoustic environment permits a relatively short coherence time with respect to the symbol period (Table 1-1), during which the channel can be reasonably assumed to be stationary [98]. This is referred to as the block-fading assumption in the communications regime [7]. (Actually, the block-fading model assumes a block-wise independent channel, while in an empirical acoustic environment the channels between successive blocks could be correlated. In particular, the effectiveness of single-carrier UAC relies heavily on such correlations, as we will show in this dissertation.)

1.2 Existing UAC Schemes

Dating back to the 1970s, early research attempts employed analog systems, essentially sophisticated loudspeakers, to explore the applicability of UAC. These preliminary UAC systems were awkward, and the resulting performance was easily affected by the underwater conditions. By taking advantage of digital modulation

over analog modulation (such as powerful error correction techniques and resistance to channel impairments), the next generation of UAC systems, featuring digital communication techniques, gained popularity in the late 1980s. Back then, with limited hardware performance and inadequate knowledge about accurate acoustic channel modeling, researchers believed that coherent UAC, such as phase-shift keying (PSK) [57], was practically unfeasible owing to the major challenges imposed by the acoustic environment (Section 1.1). As a consequence, incoherent strategies, such as frequency-shift keying (FSK) [57], drew a lot of interest instead [4, 20]. Although immune to double spreading, FSK is not a desirable modulation scheme from a spectrum efficiency point of view: at any time, only a small portion of the available bandwidth is used, leading to a data rate lower than that which would be achieved by other systems making full use of the available bandwidth. It was not until the employment of the phase locked loop (PLL) methodology [3, 29] in underwater applications that phase-coherent UWA communications became possible [14, 15, 74].

While PLL is generally successful in mitigating the effects of Doppler spreading, the delay spread can be accounted for by either the decision feedback equalizer (DFE) [56, 73] or the passive-phase conjugate (PPC) [13] methods. A detailed treatment alongside with performance comparisons of DFE and PPC is presented by Yang [94, 95]. In practical UWA systems, the coupling of DFE and PLL has found great success [73, 74] and almost became a standard [31]. In practice, the filter coefficients involved in DFE are updated by adaptive approaches such as the well-known recursive least squares (RLS) or the least mean square (LMS) algorithms [19, 30, 74]. The principle behind PPC is matched filtering: when CIR is convolved with its time-reversed and conjugated version at each receiver and added up, the summation approaches a delta function [96]. This compensates for the channel effects in the received signal. Obviously, the performance of such an approach relies heavily on the accuracy of the CIR estimate, especially when only few receivers exist. Taking one step further beyond

the classic coupling of DFE with PLL, Yang presents a hybrid structure combining the advantage of PPC with a single channel DFE [97], and introduces a Doppler shift removal module before feeding the signals to the DFE [99].

As previously remarked, the employment of MIMO schemes is leveraged by the need for achieving higher data rates in UAC. The price we have to pay is that severe interference from all other transmitters must be mitigated while detecting the signal from any given transmitter [84]. As a consequence, accurate channel estimation and symbol detection techniques, that are able to overcome the challenges of the underwater environment as well as the destructive interferences, are required. Due to its importance in increasing data rates in UAC systems, several approaches have been proposed in the literature. The minimum mean-squared error (MMSE) based linear combinatorial nulling (LCN) [11] detection scheme was considered in [66]. A MIMO reception scheme using two layers of equalization was presented in [23] and compared with the MIMO decision feedback equalizer (MIMO-DFE) [72]. A major drawback of the equalizer structure suggested in [23] is its computational complexity. A spatial modulation scheme, which assumes that accurate channel estimates are readily available at the transmitter side, was presented in [32]. In most underwater environments, such a scheme is though difficult to implement primarily since the channel characteristics vary too rapidly to allow for feedback information. When channel coding [57] is used, iterative equalization techniques, for instance, Turbo equalization [12], can be used to achieve good performance by exchanging soft information between equalizers and decoders [53, 59]. However, iterative equalization techniques also suffer from high computational complexity. As an alternative to the aforementioned time domain processing methods, [36] presents a frequency domain orthogonal frequency-division multiplexing (OFDM) approach for MIMO UAC purposes. OFDM systems are, however, generally not preferable from an amplifier efficiency point of view due to high peak-to-average-power ratio [51] and unimodular (unit modulus) sequences

are favored instead. Furthermore, the highly time-varying nature of the UAC channel poses inter-channel interference issues in multi-carrier approaches.

This dissertation focuses on single-carrier transmission scheme. We provide thorough investigation of a MIMO UAC system by providing a detailed treatment of every step involved from data transmission at the transmitter to symbol detection at the receiver. This is done by presenting approaches for designing well-structured training sequence set, effective channel estimation methods and efficient detection schemes. Both simulation and experimental results validate the utility of the proposed overall scheme for MIMO UAC.

1.3 Dissertation Outline

In Chapter 2, two key issues regarding the design of a MIMO UAC system, namely channel estimation and symbol detection, are addressed. To enhance channel estimation performance, a cyclic approach for designing training sequence set and a channel estimation algorithm called the iterative adaptive approach (IAA) are presented. Sparse channel estimates can be obtained by combining IAA with the Bayesian information criterion (BIC). Moreover, the RELAX algorithm can be used to improve the IAA with BIC estimates further. Regarding symbol detection, a linear MMSE based detection scheme, called RELAX-BLAST, which is a combination of vertical Bell Labs layered space-time (V-BLAST) algorithm and the cyclic principle of the RELAX algorithm, is presented and it is shown that RELAX-BLAST outperforms V-BLAST. Both simulated and RACE08 experimental results are provided to validate the proposed MIMO scheme.

Chapter 3 addresses the overall efficiency of the MIMO UAC reception schemes. Specifically, an efficient user parameter free Bayesian approach, referred to as sparse learning via iterative minimization (SLIM), is presented. SLIM provides good channel estimation performance along with reduced computational complexity compared to IAA. Moreover, RELAX-BLAST is implemented efficiently by making use of the

conjugate gradient method and diagonalization properties of circulant matrices. The proposed algorithm requires only simple fast Fourier transform operations and facilitates parallel implementations. SPACE08 experimental results show that the proposed MIMO UAC schemes can enjoy almost error-free performance even under severe ocean environments.

The MIMO UAC schemes presented in Chapters 2 and 3 are developed for ISI acoustic channels, i.e., the underlying channels are assumed to spread in the time domain but not in the frequency domain (Section 1.1). In Chapter 4, we incorporate Doppler effects by dealing with MIMO UAC over sparse acoustic channels suffering from both ISI and frequency modulations, e.g., motion-induced Doppler shifts. An extension of SLIM is presented to estimate the sparse and frequency modulated acoustic channels. The extended algorithm is referred to as generalization of SLIM (GoSLIM). Moreover, Chapter 4 considers channel equalization and symbol detection for various MIMO transmission schemes, including both space-time block coding and spatial multiplexing, under the challenging channel conditions. The effectiveness of the proposed approaches is demonstrated using in-water experimental measurements recently acquired during WHOI09 and ACOMM10 experiments.

Chapter 5 elaborates the future work. In this chapter, our focus is shifted from research idea development to concrete system implementation by criticizing the existing MIMO UAC schemes from an application point of view. Moreover, we also provide a vision for the future of UAC by discussing the possibilities and challenges of employing multiuser techniques in the underwater environments.

1.4 Notation

We herein list the mathematical notation used throughout this dissertation. $\|\cdot\|_2$ denotes the Euclidean norm of a vector, $||\cdot||_{\mathrm{F}}$ denotes the Frobenius matrix norm, and $|\cdot|$ is the modulus. $(\cdot)^*$ is the complex conjugate of a scalar, $(\cdot)^T$ and $(\cdot)^H$ denote the transpose and conjugate transpose, respectively, of a matrix or vector. $E(\cdot)$ denotes the

expected value, $\mathbf{I}$ denotes the identity matrix of appropriate dimension and $\hat{\mathbf{x}}$ denotes the estimate of $\mathbf{x}$. $\mathrm{diag}(\mathbf{v})$ represents a diagonal matrix in which the elements of $\mathbf{v}$ are on the diagonal. The ith column of a matrix $\mathbf{X}$ is written as $\mathbf{x}_i$. Re $(\cdot)$ represents the real component of a complex-valued vector of matrix. Other mathematical symbols are defined after their first appearance.

Figure 1-1. An unmanned underwater vehicle (UUV). Copyright image courtesy of Woods Hole Oceanographic Institution.

A B

Figure 1-2. A) The burning Deepwater Horizon oil platform after a massive explosion in the Gulf of Mexico on April 20, 2010. B) The Gulf of Mexico on May 6, 2010. Copyright images courtesy of the New York Times.

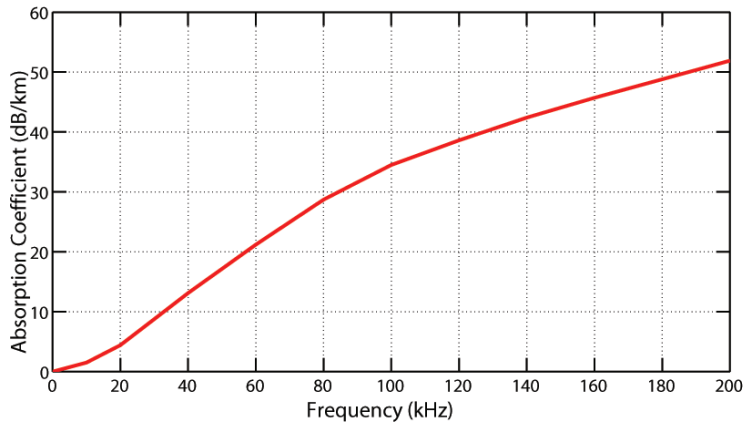


Figure 1-3. Absorption coefficient versus frequency. Copyright image courtesy of [76].

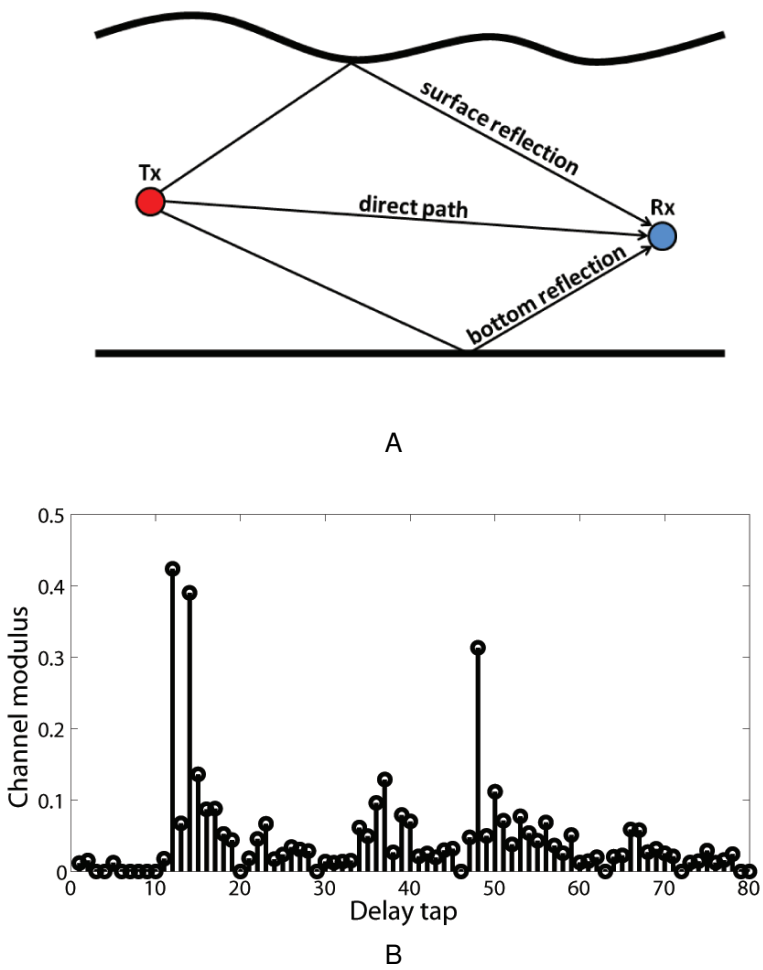
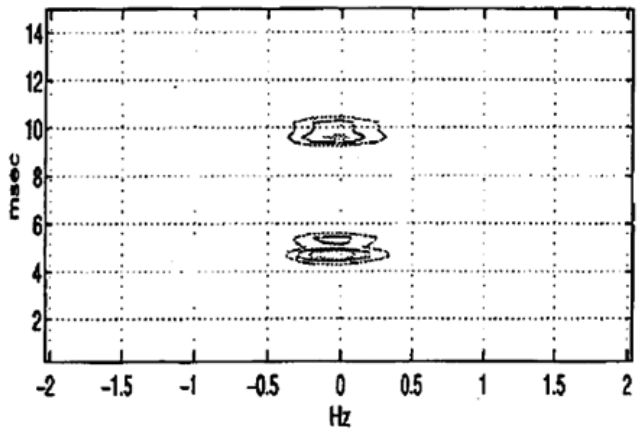
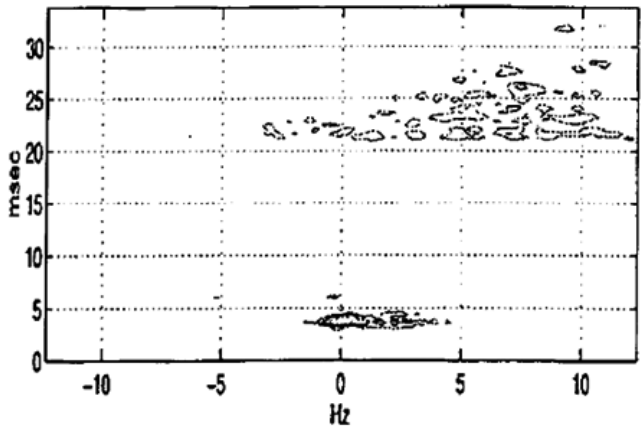


Figure 1-4. A) An underwater acoustic channel with 3 multipaths. B) A practical channel impulse response (CIR) with 80 channel taps.



A



B

Figure 1-5. Scattering functions obtained at two different underwater acoustic conditions. A) Arctic environment with frozen sea surface. B) Bahama Islands on a windy day. The contours are in 3-dB increments. Copyright image courtesy of [32].

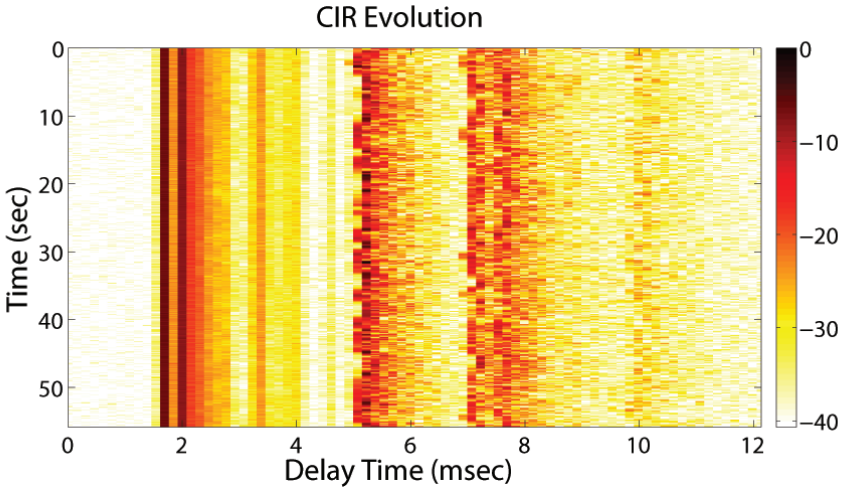


Figure 1-6. Normalized CIR evolution over approximately a 1 min period.

CHAPTER 2
ENHANCED CHANNEL ESTIMATION AND SYMBOL DETECTION FOR HIGH SPEED MULTI-INPUT MULTI-OUTPUT UAC

This chapter focuses on the various aspects of using a MIMO acoustic communications system in an underwater environment where delay spread is present. The problem was divided into two main parts: *i)* channel estimation, which involves the design of the training sequences and the design of the algorithm to estimate the channel coefficients using the training sequences or previously detected symbols and *ii)* symbol detection. In general, the very first task of the receiver is to conduct a training-directed channel estimation [43, 74]. To achieve good performance, both well-structured training sequences and a signal processing methodology that can estimate the CIR accurately using the designed training sequences are required. In addition, to address the time-varying nature of the underwater acoustic channel, the decision-directed channel estimation is performed regularly using the detected symbols [43, 74]. Therefore, the channel estimation algorithm should be able to work well both in training- and decision-directed modes.

For practical ISI channels encountered in UAC, sequences with good auto- and cross-correlation properties instead are required [90, 93]. Early research has focused on binary training sequences [16, 93] due to practical concerns and simplicity. Later on, the use of polyphase training sequences was proposed, where the possible phase values were confined to a predefined finite set [90]. It is obviously advantageous to allow the phase values to be continuous. The cyclic approach (CA) presented by Li et. al. [40, 41] for probing sequence design enjoys superior performance over the aforementioned methods by allowing continuous phase values while still being computationally tractable. The training sequences designed using the CA methodology possess good auto- and cross-correlation properties as desired for MIMO channel estimation in communications [40, 41].

The second phase of channel estimation involves the design of the algorithm that will estimate the CIR using the training sequences (or the previously detected symbols) and the received measurements. To address the sparsity of the acoustic channel [5, 33, 73, 75], three important sparsity based techniques have been used for underwater channel estimation, namely the matching pursuit (MP) algorithm, the orthogonal MP (OMP) algorithm [52], and the least squares MP algorithm (LSMP) [8–10, 42, 43, 50, 52]. These methods, however, involve user parameter that is difficult to determine, and their performance might degrade significantly depending on the structure of the matrix relating the unknowns to the measurements. To address these problems, we present a user parameter-free nonparametric iterative adaptive approach (IAA) [100] for estimating the CIR accurately even when the training sequences are arbitrary and short in length. The dominant channel tap estimates of IAA can be used in a Bayesian information criterion (BIC) [63, 70] to decide which taps to retain and which ones to discard. This combined method, called IAA with BIC, results in sparse channel estimates. Further improvements in performance can be achieved by initializing the last step of the cyclic and relaxation-based RELAX [38, 39] algorithm via the IAA with BIC sparse estimates.

Following the estimation of the CIR is the design of the detection scheme for extracting the payload symbols from the measurements. We use a linear minimum mean-squared error (LMMSE) based filter for signal detection. Two important methods for applying the LMMSE filter coefficients to the measurements are the linear combinatorial nulling [11] and vertical Bell Labs Layered Space-Time (V-BLAST) algorithms [92]. It is interesting to note that these two approaches resemble the classical periodogram [69, 88] and the CLEAN [27] methods used in spectral estimation applications. Being inspired from the improvements of RELAX over the periodogram and CLEAN [69], we propose the RELAX-BLAST detection algorithm, which is a combination of V-BLAST

and the cyclic principle of RELAX as the name suggests, and show that it outperforms V-BLAST.

The rest of this chapter is organized as follows. Section 2.1 outlines the system configuration and describes the data package structure. Section 2.2 formulates the problem of CIR estimation, describes the CA method for training sequence design and presents the IAA algorithm together with the BIC and RELAX extensions. Next, the symbol detection problem is analyzed in Section 2.3 and the LMMSE based RELAX-BLAST detection scheme is proposed. Both simulated and experimental results are presented in Section 2.4. The sea data was gathered in the rescheduled acoustic communications experiment (RACE08), which was conducted by the Woods Hole Oceanographic Institution (WHOI) in Narragansett Bay.

2.1 System Outline

Consider an $N \times M$ MIMO UAC system equipped with N transmit transducers and M receive transducers. The individual data streams of each transmitter are symbol aligned and are sent simultaneously. The data streams of each transmitter consist of successive data packages of the form shown in Figure 2-1. The data packages start with a training sequence of length P which is followed by a silent gap, the payload sequence and another silent gap. During the gap intervals, no signal is transmitted in order to prevent the inner-package ISI (Gap 1) between the training and payload symbols and the inter-package ISI (Gap 2) between two consecutive packages. The payload sequence, which has length Q ($Q > P$ in general), is the estimation target and each payload symbol is drawn from a quadrature PSK (QPSK) constellation modulated with Gray code [57]. The four constellation points of the QPSK symbols, i.e., $\{e^{j(2n-1)\frac{\pi}{4}}\}_{n=1}^{4}$, lie on the unit circle. Such a constellation is desirable in practice due to its unit modulus. The same practical constraints require the training symbols to have unit modulus as well but no restriction is imposed on their phase values.

In what follows, our consideration is always confined to one data package of the form given in Figure 2-1. Let $x_n(t)$ denote the tth symbol in the package sent by the nth transmitter and let $y_m(t)$ denote the tth symbol in the package received by the mth receiver, where $n = 1, \ldots, N$, $m = 1, \ldots, M$, $t = 1, \ldots, T$, and T is the total symbol length of a single transmitted package. We do not go into the details of the sampling and synchronization procedures and assume that such operations have already been employed and the sampled complex baseband signals are available at the receiver.

Figure 2-2 shows the $N \times M$ MIMO system structure that we will use throughout the chapter. The source bits are encoded, QPSK modulated, interleaved and demultiplexed for transmission from multiple transducers. A random interleaver is used in order to avoid burst errors, which occur when the channel behaves badly at certain intervals of time [57]. After the signals have been received by the receive array, the processing consists of two steps: estimating the CIR (in training- or decision-directed mode) and detecting the symbols by using the estimated CIR. Once the symbols have been detected, they are multiplexed, deinterleaved and then fed into a Viterbi decoder to recover the source bits. We now discuss the channel estimation problem.

2.2 Channel Estimation

2.2.1 Problem Formulation

In the training-directed mode, an initial CIR estimate is obtained by using the training sequences sent at the beginning of each package whereas in the decision-directed mode, the previously detected symbols are used to update the most recent CIR estimate. How frequently the channel estimate has to be updated depends on the channel characteristics. For a nonstationary channel (relative to the total length of the first gap and the payload sequence), the CIR has to be updated frequently whereas for a stationary channel, the initial CIR estimate might yield sufficient performance for detecting the entire payload sequence.

2.2.1.1 Training-directed mode

The measurement vector at the mth receiver can be written as:

$$\mathbf{y}_m = \sum_{n=1}^{N} \tilde{\mathbf{X}}_n \mathbf{h}_{n,m} + \mathbf{e}_m \tag{2–1}$$

for $m = 1, \ldots, M$, where

$$\mathbf{y}_m = [y_m(1), \ldots, y_m(P+R-1)]^T, \tag{2–2}$$

$$\mathbf{h}_{n,m} = [h_{n,m}(1), \ldots, h_{n,m}(R)]^T, \tag{2–3}$$

The vector $\mathbf{h}_{n,m}$ here denotes the CIR between the nth transmitter and the mth receiver and it contains R channel taps,

$$\tilde{\mathbf{X}}_n = \begin{bmatrix} x_n(1) & \ldots & \mathbf{0} \\ \vdots & \ddots & \\ x_n(P) & & x_n(1) \\ & \ddots & \vdots \\ \mathbf{0} & \ldots & x_n(P) \end{bmatrix}, \tag{2–4}$$

where $\tilde{\mathbf{X}}_n \in \mathbb{C}^{(P+R-1)\times R}$ contains the nth training sequence and hence is known, and $\mathbf{e}_m$ is additive noise (thermal or hardware related noise) at the mth receiver. (2–1) can be rewritten as:

$$\mathbf{y}_m = \mathbf{X}\mathbf{h}_m + \mathbf{e}_m, \tag{2–5}$$

where $\mathbf{X} = [\tilde{\mathbf{X}}_1 \cdots \tilde{\mathbf{X}}_N]$ and $\mathbf{h}_m = [\mathbf{h}_{1,m}^T \cdots \mathbf{h}_{N,m}^T]^T$. The training-directed channel estimation problem then reduces to estimating $\mathbf{h}_m$ from the measurements $\mathbf{y}_m$ and known $\mathbf{X}$. It is assumed that the channel is stationary over the length of $\mathbf{y}_m$. In order to estimate all the channels for the $N \times M$ MIMO system, (2–5) has to be solved for $m = 1, \ldots, M$, i.e., M times. Note that $\mathbf{X}$ does not depend on m.

2.2.1.2 Decision-directed mode

The problem in the decision-directed mode is very similar to that of the training-directed mode except that now the training symbols are replaced with the previously estimated payload symbols. Accordingly, (2–1) and (2–5) can still be used, where

$$\mathbf{y}_m = [y_m(t_i), \dots, y_m(t_f)]^T, \quad m = 1, \dots, M, \tag{2–6}$$

contains the measurements at the mth receiver belong to the time index interval $[t_i, t_f]$, and

$$\tilde{\mathbf{X}}_n = \begin{bmatrix} \hat{x}_n(t_i) & \hat{x}_n(t_i - 1) & \dots & \hat{x}_n(t_i - R + 1) \\ \hat{x}_n(t_i + 1) & \hat{x}_n(t_i) & \dots & \hat{x}_n(t_i - R + 2) \\ \vdots & \vdots & & \vdots \\ \hat{x}_n(t_f) & \hat{x}_n(t_f - 1) & \dots & \hat{x}_n(t_f - R + 1) \end{bmatrix}, \quad n = 1, \dots, N, \tag{2–7}$$

where $\hat{x}_n(t_i - R + 1)$ and $\hat{x}_n(t_f)$ represent the first and the last previously estimated symbols (some of them could be the known training symbols), respectively, used for updating the channel. (For notational simplicity, $\tilde{\mathbf{X}}_n$ is used in (2–4) and (2–7) to represent two similar but different quantities, which should be clear from the context.) The decision-directed channel estimation problem reduces to estimating $\mathbf{h}_m$ from the measurements $\mathbf{y}_m$ and the previously decoded symbols in $\mathbf{X}$. On the one hand, it would be beneficial to keep $L \triangleq t_f - t_i + 1$ (i.e., the number of rows of $\tilde{\mathbf{X}}_n$) large for estimating the channel more accurately but on the other hand, for a rapidly varying channel, L must be kept small so that the stationarity assumption of the channel over the length of $\mathbf{y}_m$ holds and so that the channel can be updated more frequently. Therefore, L is a trade-off parameter which should be set according to the experimental conditions.

2.2.2 Training Sequence Design

We use the algorithm presented in [40, 41] for designing training sequences such that $\mathbf{X}$ in (2–5) facilitates the estimation of the CIR. It is desirable to have training symbols with constant modulus, i.e., the training symbols should have the following

generic form:

$$x_n(t) = e^{j\phi_n(t)}, \;\; t = 1, \dots, P, \;\; n = 1, \dots, N, \tag{2–8}$$

where $\phi_n(t) \in [0, 2\pi)$ represents the phase of the tth training symbol sent by the nth transmitter. Ideally, if $\mathbf{X}^H\mathbf{X} = P\mathbf{I}$ (called the pairwise orthogonality principle), then the channel estimates can be recovered perfectly by matched filtering in the noiseless case. However, pairwise orthogonality is hardly achievable, if not impossible, in practice [41]. Instead, $\epsilon = \|\mathbf{X}^H\mathbf{X} - P\mathbf{I}\|_{\mathrm{F}}^2$ can be made small.

Let $\mathbf{U}$ be an arbitrary semi-unitary matrix (i.e., $\mathbf{U}\mathbf{U}^H = \mathbf{I}$). Then,

$$\epsilon = \|\mathbf{X}^H\mathbf{X} - (\sqrt{P}\mathbf{U})(\sqrt{P}\mathbf{U}^H)\|_{\mathrm{F}}^2. \tag{2–9}$$

Minimizing ϵ can then be formulated in the following related (but not equivalent) way [41]:

$$\{\phi_n(t)\} = \arg \min_{\{\phi_n(t)\}, \mathbf{U}^H} \|\mathbf{X} - \sqrt{P}\mathbf{U}^H\|_{\mathrm{F}}^2, \quad \text{s.t.} \quad \mathbf{U}\mathbf{U}^H = \mathbf{I}. \tag{2–10}$$

This optimization problem can be solved efficiently by using the CA method [41, 81] which guarantees that the cost function does not increase as the iterations proceed. In the CA method, $\mathbf{U}$ is assumed given when estimating $\{\phi_n(t)\}$ and vice versa. This way, the optimization problem is solved iteratively by dividing it into simpler sub-problems. When $\mathbf{U}^H$ is fixed, the solution to (2–10) has the generic form:

$$\phi = \arg\left(\sum_{r=1}^{R} z_r\right). \tag{2–11}$$

where $\{z_r\}_{r=1}^{R}$ are given numbers. For example, when the update target is $\phi_1(1)$, z_r represents the (r, r)th diagonal entry of $\sqrt{P}\mathbf{U}^H$. Given the phases $\phi_n(t)$, the solution to (2–10) is given by $\mathbf{U}^H = \bar{\mathbf{U}}\tilde{\mathbf{U}}^H$ [28, 41], where

$$\sqrt{P}\mathbf{X} = \bar{\mathbf{U}}\mathbf{\Gamma}\tilde{\mathbf{U}}^H \tag{2–12}$$

is the singular value decomposition (SVD) of $\sqrt{P}\mathbf{X}$ ($\bar{\mathbf{U}}$ and $\tilde{\mathbf{U}}^H$ are unitary matrices and $\mathbf{\Gamma}$ is a diagonal matrix with the singular values of $\sqrt{P}\mathbf{X}$ on its diagonal).

The CA algorithm is terminated when the difference of the cost function (defined in (2–10)) between two sucessive iterations drops below a certain threshold. For the CA algorithm to show good performance, it is recommended that $P \gg R$ and $NR < P+R-1$ [40, 41]. In practice, N is determined from the system configuration while R is selected depending on the experimental conditions and is expected to be the smallest value that can capture the prominent channel features. It seems as if a large P value is preferable for satisfying the two inequalities. However, there are two problems associated with increasing P. First, the accuracy of the initial channel estimation depends on the assumption that the channel is stationary. For a fixed symbol rate, larger P means longer transmission time which means the stationarity assumption is more likely to be violated. Secondly, larger P means larger overhead and hence lower net data rate. Fortunately, though, the two inequalities can in general be satisfied in practice by selecting the parameters appropriately. Note that the CA method has been used to design the training sequences in the numerical and experimental results section of this chapter.

2.2.3 Channel Estimation Algorithm

The channel estimation problem at each receiver, in either training-directed mode or decision-directed mode per the discussion in 2.2.1, has the generic form given by:

$$\mathbf{y} = \mathbf{X}\mathbf{h} + \mathbf{e}, \tag{2–13}$$

where we have omitted the index m for notational simplicity. Note that the number of elements in $\mathbf{y}$, namely d_y, might be different for the two modes. The problem is then to estimate $\mathbf{h}$, which has NR unknowns, given $\mathbf{y}$ and $\mathbf{X}$. In the following, we present the IAA algorithm [100] to solve this problem. IAA makes no assumptions on the statistical properties of the additive noise $\mathbf{e}$. Note that since $\mathbf{h}$ contains the CIR of all N transmitters, IAA will estimate them jointly.

2.2.3.1 Iterative adaptive approach (IAA)

Many existing weighted least squares (WLS) based channel estimation methodologies require the tuning of one or more user parameters and their assumptions on the CIR are in general not valid in the underwater scenario [34, 83]. To account for these problems, we present a user parameter-free iterative WLS based channel estimation technique, called IAA [100]. IAA is an adaptive and nonparametric algorithm, and it does not make any explicit assumptions on the CIR. Let $\mathbf{P}$ be an $NR \times NR$ diagonal matrix whose diagonal contains the squared absolute value of each channel tap, i.e.,

$$P_r = |h_r|^2, \quad r = 1, \dots, NR. \tag{2–14}$$

where P_r is the rth diagonal element of $\mathbf{P}$ and h_r is the rth element of $\mathbf{h}$. The covariance matrix of the noise and interference with respect to the tap of current interest h_r can be expressed as:

$$\mathbf{Q}(r) = \mathbf{R} - P_r \mathbf{x}_r \mathbf{x}_r^H, \tag{2–15}$$

where $\mathbf{R} \triangleq \mathbf{XPX}^H$. Then, the WLS cost function is given by [37, 67–69]:

$$(\mathbf{y} - h_r \mathbf{x}_r)^H \mathbf{Q}^{-1}(r) (\mathbf{y} - h_r \mathbf{x}_r). \tag{2–16}$$

Minimizing (2–16) with respect to h_r yields

$$\hat{h}_r = \frac{\mathbf{x}_r^H \mathbf{Q}^{-1}(r) \mathbf{y}}{\mathbf{x}_r^H \mathbf{Q}^{-1}(r) \mathbf{x}_r}. \tag{2–17}$$

Using (2–15) and the matrix inversion lemma, (2–17) can be written as

$$\hat{h}_r = \frac{\mathbf{x}_r^H \mathbf{R}^{-1} \mathbf{y}}{\mathbf{x}_r^H \mathbf{R}^{-1} \mathbf{x}_r}. \tag{2–18}$$

This avoids the computation of $\mathbf{Q}^{-1}(r)$ for NR times and only one matrix inversion is needed per iteration. IAA for channel estimation is summarized in Table 2-1. Since IAA requires $\mathbf{R}$, which itself depends on the unknown channel taps, it has to be implemented as an iterative approach. The initialization is done by a standard matched

Table 2-1. Iterative adaptive approach (IAA)

$P_r = \frac{\lvert \mathbf{x}_r^H \mathbf{y} \rvert^2}{(\mathbf{x}_r^H \mathbf{x}_r)^2}, \quad r = 1, 2, \ldots, NR$
repeat
$\mathbf{R} = \mathbf{X}\mathbf{P}\mathbf{X}^H$
$\hat{h}_r = \frac{\mathbf{x}_r^H \mathbf{R}^{-1} \mathbf{y}}{\mathbf{x}_r^H \mathbf{R}^{-1} \mathbf{x}_r}, \quad r = 1, 2, \ldots, NR$
$P_r = \lvert \hat{h}_r \rvert^2, \quad r = 1, 2, \ldots, NR$
until (convergence)

filter. Our empirical experience is that IAA does not provide significant improvements in performance after about 15 iterations. In IAA, $\mathbf{P}$ and hence $\mathbf{R}$ are obtained from the channel estimates of the previous iteration and not from the measurements $\mathbf{y}$ as done in conventional adaptive filtering algorithms.

If the computation of $\mathbf{R}$ becomes problematic due to numerical ill-conditioning during the iterations, a regularization approach can be used. IAA can be regularized by considering an additional noise term separately from the interference terms in the expression for $\mathbf{R}$:

$$\mathbf{R} = \mathbf{X}\mathbf{P}\mathbf{X}^H + \mathbf{\Sigma}, \tag{2–19}$$

where $\mathbf{\Sigma}$ is a diagonal matrix with unknown noise powers $\{\sigma_m^2\}_{m=1}^{d_y}$ on its diagonal. IAA is then implemented in the same way as before except that now there are $NR + d_y$ rather than NR unknowns. Consequently, $\{\sigma_m^2\}$ can be estimated by

$$\hat{\sigma}_m^2 = \frac{\lvert \mathbf{i}_m^H \mathbf{R}^{-1} \mathbf{y} \rvert^2}{(\mathbf{i}_m^H \mathbf{R}^{-1} \mathbf{i}_m)^2}, \quad m = 1, \ldots, d_y, \tag{2–20}$$

at each iteration, where $\mathbf{i}_m$ is the mth column of the $d_y \times d_y$ identity matrix. Since the diagonal loading levels are calculated automatically, the approach conserves the practicality of IAA. Setting $\{\hat{\sigma}_m\}_{m=1}^{d_y}$ to zero gives the original IAA algorithm. $\mathbf{\Sigma}$ can be initialized as all zeros.

Table 2-2. IAA with the Bayesian information criterion.

$\mathcal{P} = \{1, \ldots, NR\}$
$\mathcal{I} = \{\emptyset\}$; $\eta = 1$; quit = 0; BIC$^{old} = \infty$
repeat
$\tilde{i} = \arg\min_{i \in \mathcal{P} - \mathcal{I}} \text{BIC}_i(\eta)$
if BIC$_{\tilde{i}}(\eta)$ <BIC^{old}
$\mathcal{I} = \{\mathcal{I}, \tilde{i}\}$
BIC^{old} =BIC$_{\tilde{i}}(\eta)$
$\eta = \eta + 1$
else quit $= 1$
until (quit $= 1$)

2.2.3.2 IAA with the Bayesian information criterion

In order to achieve sparsity with IAA, i.e., to retain only a few dominant channel taps, the Bayesian information criterion (BIC) [63, 70], can be used in conjunction with IAA. BIC is a model order selection tool that is widely used in the statistics and signal processing communities. The advantage of using BIC over a simple thresholding operation is that BIC does not require the manual specification of a threshold value. (Note that the selection of the threshold value has a significant effect on the overall performance and it is usually impractical to tune this value for best performance since the true CIR is unknown.) Let $\mathcal{P}$ denote a set containing the indices of all the channel taps. Also, let $\mathcal{I}$ denote the set of the indices of the taps selected by the BIC algorithm so far. The IAA with BIC algorithm works as follows: first, the tap from the set $\mathcal{P}$ giving the minimum BIC is selected and included in the set $\mathcal{I}$ (initially $\mathcal{I} = \emptyset$). Then the second tap, from the set $\mathcal{P} - \mathcal{I}$, which together with the first tap gives the minimum BIC is selected and so on until the BIC value does not decrease anymore. The IAA with BIC algorithm is summarized in Table 2-2. BIC$_i(\eta)$ is calculated as follows [70],

$$\text{BIC}_i(\eta) = 2d_y \ln\left(||\mathbf{y} - \sum_{j \in \{\mathcal{I} \cup i\}} \mathbf{x}_j \hat{h}_j||_2^2\right) + 1.5\eta \ln(2d_y), \tag{2–21}$$

Table 2-3. IAA with RELAX.

$\mathcal{I}$: Indices of the taps selected by IAA with BIC
$K = |\mathcal{I}|$, i.e., the number of selected taps
repeat
  for $k = 1, 2, \ldots, K$
    $\mathbf{y}_k = \mathbf{y} - \sum_{i=1, i\neq k}^{K} \mathbf{x}_{\mathcal{I}(i)} \hat{h}_{\mathcal{I}(i)}$
    $\hat{h}_{\mathcal{I}(k)} = \mathbf{x}_{\mathcal{I}(k)}^H \mathbf{y}_k / \|\mathbf{x}_{\mathcal{I}(k)}\|_2^2$
  end for
until (convergence)

---

where $\eta = |\mathcal{I}| + 1$, with $|\mathcal{I}|$ denoting the size of $\mathcal{I}$, i is the index of the current tap under consideration, and $\hat{h}_j$ is the IAA estimate of the jth element of $\mathbf{h}$, $j \in \{\mathcal{I} \cup i\}$. After BIC is implemented, the indices of the surviving CIR taps can be found in $\mathcal{I}$. All other channel taps are then set to zero.

2.2.3.3 IAA with RELAX

The parametric and cyclic RELAX algorithm [38, 39] which was originally proposed for spectral estimation, can be used to improve the IAA with BIC results even further. Because RELAX is parametric, it requires the number of sources to be known. The IAA with BIC result can be used to estimate the number of sources and also to provide initial estimates for the last step of RELAX as shown in Table 2-3. Note that $\mathcal{I}(k)$ denotes the kth element in the set $\mathcal{I}$. The idea presented in Table 2-3 is to remove the contribution from all the components of $\hat{\mathbf{h}}$ other than the one of current interest $\hat{h}_{\mathcal{I}(k)}$ and then to update $\hat{h}_{\mathcal{I}(k)}$ in the minimum least squares sense. This procedure is repeated until the difference of the cost function $\|\mathbf{y} - \mathbf{X}\hat{\mathbf{h}}\|_2^2$ between two successive iterations becomes less than a certain threshold. (We used a threshold of 5×10^{-3} in our simulations herein.) For the best performance, it is recommended that before each RELAX iteration, $\{\hat{h}_k\}$ be sorted by their magnitude in descending order and the columns of $\mathbf{X}$ be permuted accordingly. This way, the tap with the largest magnitude will be updated first, the tap with the second largest magnitude will be updated next and so on.

2.2.3.4 Complexity analysis

The initialization step of IAA has complexity $\mathcal{O}(2d_y(NR) + 3(NR))$ and each IAA iteration has complexity $\mathcal{O}(d_y^3 + (2d_y^2 + 3d_y + 2)(NR))$. These complexities are calculated by counting the multiplication and division operations in Table 2-1. When $d_y > (NR)$, $\mathbf{R}^{-1}$ can be calculated only once at the initialization step of IAA and then it can be updated when every $\{P_r\}$ is estimated using the rank-1 matrix inverse update formula [22]. This way, the complexity of computing $\mathbf{R}^{-1}$ reduces to $\mathcal{O}((d_y^2 + 3)(NR))$ rather than $\mathcal{O}(d_y^3)$ at each IAA iteration. The resulting complexity of IAA is then given by $\mathcal{O}(d_y^3 + (d_y^2 + 3d_y + 3)(NR))$ for initialization and $\mathcal{O}((2d_y^2 + 2d_y + 5)(NR))$ per IAA iteration. The complexity of IAA is smaller than those of MP and LSMP when $d_y \ll (NR)$ and larger when $d_y > (NR)$ [43]. However, the computation time does not depend only on the number of computations but rather is a function of the memory access time, the implementation software and hardware and the number of computations combined together. Note that the regularization, BIC and RELAX extensions will be applied in all of our numerical examples and henceforth this combined approach will simply be referred to as IAA.

2.3 Symbol Detection

2.3.1 Problem Formulation

Treating the transmitted symbols as the unknowns instead of the CIRs in (2–1), the measurement can be expressed as [43, 56]:

$$\mathbf{y}_m = \sum_{n=1}^{N} \hat{\mathbf{H}}_{n,m} \dot{\mathbf{x}}_n + \mathbf{e}_m, \quad m = 1, \dots, M, \tag{2–22}$$

where $\hat{\mathbf{H}}_{n,m} \in \mathbb{C}^{R \times (2R-1)}$ is given by

$$\hat{\mathbf{H}}_{n,m} = \begin{bmatrix} \hat{h}_{n,m}(R) & \dots & \hat{h}_{n,m}(1) & & \mathbf{0} \\ \vdots & \ddots & & \ddots & \vdots \\ \mathbf{0} & & \hat{h}_{n,m}(R) & \dots & \hat{h}_{n,m}(1) \end{bmatrix}, \quad n = 1, \dots, N, \quad m = 1, \dots, M, \tag{2–23}$$

$$\begin{aligned}\dot{\mathbf{x}}_n &= [x_n(t_0-R+1),\dots,x_n(t_0),\dots,x_n(t_0+R-1)]^T, \quad n=1,\dots,N,\\ \mathbf{y}_m &= [y_m(t_0),\dots,y_m(t_0+R-1)]^T, \quad m=1,\dots,M,\end{aligned} \tag{2–24}$$

and t_0 represents the time index corresponding to the payload symbol of interest. Note that $\mathbf{y}_m$ represents the measurements in (2–2), (2–6) and (2–24) since $\mathbf{y}_m$ represents a portion of the received signal in any case. However, the use of $\mathbf{y}_m$ should be clear from the context. Stacking up all the measurements, (2–22) can be written as

$$\begin{bmatrix}\mathbf{y}_1\\ \vdots\\ \mathbf{y}_M\end{bmatrix} = \sum_{n=1}^{N}\begin{bmatrix}\hat{\mathbf{H}}_{n,1}\\ \vdots\\ \hat{\mathbf{H}}_{n,M}\end{bmatrix}\dot{\mathbf{x}}_n + \begin{bmatrix}\mathbf{e}_1\\ \vdots\\ \mathbf{e}_M\end{bmatrix}, \tag{2–25}$$

or, equivalently as,

$$\tilde{\mathbf{y}} = \sum_{n=1}^{N}\hat{\mathbf{H}}_n\dot{\mathbf{x}}_n + \tilde{\mathbf{e}}, \tag{2–26}$$

where $\tilde{\mathbf{y}}$ and $\tilde{\mathbf{e}} \in \mathbb{C}^{MR\times 1}$, and $\{\hat{\mathbf{H}}_n\}_{n=1}^N \in \mathbb{C}^{MR\times(2R-1)}$. The transmitted symbols $\{x_n(t_0)\}_{n=1}^N$ are estimated using $\tilde{\mathbf{y}}$ in (2–26). When estimating $\{x_n(t_0+1)\}_{n=1}^N$, the measurement vector $\tilde{\mathbf{y}}$ is shifted by one symbol duration, i.e., t_0 is replaced by t_0+1, and so on. Note that when detecting the symbols, the channel is assumed constant since the previous channel update, which allows us to treat $\{\hat{\mathbf{H}}_n\}_{n=1}^N$ in (2–26) as known.

2.3.2 The Linear Minimum Mean-Squared Error (LMMSE) Filter

In this section, we briefly review the Wiener filter [61, 91], which is optimal in the MMSE sense with respect to each transmitted symbol, for symbol detection. The Wiener filter is widely used in the communication literature [49, 54, 92] and the exposition provided in this section is for the sake of completeness. The steering vector corresponding to $\{x_n(t_0)\}$ in (2–26) is given by $\mathbf{s}_n \triangleq [\hat{\mathbf{h}}_{n,1}^T \cdots \hat{\mathbf{h}}_{n,M}^T]^T$ where $\hat{\mathbf{h}}_{n,m}$ are the estimates of $\mathbf{h}_{n,m}$ defined in (2–3). We Let the symbol of current interest be $x_n(t_0)$. Then, the Wiener filter for this symbol, denoted as $\mathbf{g}_n$, can be derived by solving:

$$\mathbf{f}_n = \arg\min_{\mathbf{f}} E(||\mathbf{f}^H\tilde{\mathbf{y}} - x_n(t_0)||_2^2). \tag{2–27}$$

The solution to (2–27) is [61, 91]:

$$\mathbf{f}_n = \mathbf{R}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}^{-1} E(x_n^H(t_0)\tilde{\mathbf{y}}), \tag{2–28}$$

where $\mathbf{R}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}$ is the covariance matrix of $\tilde{\mathbf{y}}$, i.e., $\mathbf{R}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}} = E\left(\tilde{\mathbf{y}}\tilde{\mathbf{y}}^H\right)$.

In the following, it is assumed that the payload sequences are pairwise uncorrelated, each payload sequence is uncorrelated with the noise $\tilde{\mathbf{e}}$, the noise has zero mean, each payload symbol is independent of the other payload symbols and each payload symbol has zero mean. By using these assumptions, it is easy to verify that:

$$\mathbf{R}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}} = \tilde{\mathbf{H}}\tilde{\mathbf{H}}^H + \mathbf{R}_{\tilde{\mathbf{e}}\tilde{\mathbf{e}}} \tag{2–29}$$

where $\tilde{\mathbf{H}} = [\tilde{\mathbf{H}}_1, \dots, \tilde{\mathbf{H}}_N]$ and $E\left(x_n^H(t_0)\tilde{\mathbf{y}}\right) = \mathbf{d}_n$. (2–28) then becomes:

$$\mathbf{f}_n = \left(\hat{\mathbf{H}}\hat{\mathbf{H}}^H + \mathbf{R}_{\tilde{\mathbf{e}}\tilde{\mathbf{e}}}\right)^{-1} \mathbf{s}_n \tag{2–30}$$

and the soft estimate of the symbol $x_n(t_0)$ is given by $\mathbf{f}_n^H\tilde{\mathbf{y}}$. In our experiments we estimate $\mathbf{R}_{\tilde{\mathbf{e}}\tilde{\mathbf{e}}}$ from the residual error obtained during the channel estimation process, i.e., using $\mathbf{e}_m = \mathbf{y}_m - \mathbf{X}\hat{\mathbf{h}}_m$, $m = 1, \dots, M$, in (2–13). Since digital communications require the receiver to make a hard decision, the nearest constellation point to $\mathbf{f}_n^H\tilde{\mathbf{y}}$ is selected as the symbol estimate.

2.3.3 Detection Schemes

In the following, we will consider three approaches for applying the filters $\{\mathbf{f}_n\}$ to the measurements. We will note the relations between the approaches proposed in the communications literature with those in the spectral estimation area and propose a new scheme inspired by this relationship.

2.3.3.1 Linear combinatorial nulling

In linear combinatorial nulling (LCN) [11], $x_n(t_0)$ is detected using $\mathbf{f}_n^H\tilde{\mathbf{y}}$ for $n = 1, \dots, N$ separately where for each n, other symbols are simply treated as interferences, i.e., the estimation of $x_n(t_0)$ has no effect on the estimation of $x_{\tilde{n}}(t_0)$ ($n \neq \tilde{n}$). However,

this approach shows poor performance when the channel coefficients for each transmitter differ significantly in magnitude. For instance, when the channel coefficients of the first transmitter dominate all the others, the symbol estimate for the first transmitter will be relatively accurate whereas the symbols sent from the other transmitters will be buried under the contribution from the first transmitter and hence they will be estimated inaccurately.

2.3.3.2 CLEAN-BLAST

The idea of sequential cancellation and nulling (SCN) can be used to alleviate the aforementioned drawback of LCN. As the name implies, SCN first detects the symbol with the strongest channel response. Then, the contribution of this symbol is removed from the measurements $\tilde{\mathbf{y}}$ (and the corresponding columns are removed from $\tilde{\mathbf{H}}$) before estimating the other symbols. This process continues until all the N symbols are estimated. The symbol with the strongest channel coefficients is detected first because it can be estimated more accurately than the other symbols with weaker channel coefficients. After the dominant symbols are subtracted from the measurements, the weaker symbols can be estimated more accurately. Sequential cancellation, from the viewpoint of the remaining symbols, can be recognized as interference removal. Eventually, when detecting the symbol with the weakest channel coefficients, no more interferences are present. The detection algorithm featuring SCN is called V-BLAST [92]. Herein, we name the algorithm as CLEAN-BLAST to emphasize its analogy to the CLEAN algorithm used in spectral estimation [64].

2.3.3.3 RELAX-BLAST

As we have already pointed out, the relationship between LCN and CLEAN-BLAST is analogous to that of the periodogram and CLEAN [69]. In spectral estimation, RELAX is also called SUPER-CLEAN [38, 39] since it is a recursive version of CLEAN but with much better performance. Both V-BLAST and RELAX-BLAST take advantage of sequential interference cancellation (SIC) [84] To achieve satisfactory detection

performance, the detection order should be chosen carefully in accordance with the received signal power. The detection order is determined as follows. Once $\{\hat{\mathbf{h}}_{n,m}\}$ is available for $n = 1, \ldots, N$ and $m = 1, \ldots, M$, the received power for the nth transmitted stream, denoted as $\gamma(n)$, is calculated as $\gamma(n) = \sum_{m=1}^{M} \|\hat{\mathbf{h}}_{n,m}\|^2$. The receiver first detects the stream associated with the strongest power among $\{\gamma(n)\}_{n=1}^{N}$, followed by the detection of the stream associated with the second strongest value and so on.

In the same spirit as RELAX, RELAX-BLAST first detects the symbol with the dominant channel taps and subtracts it out from $\tilde{\mathbf{y}}$. Then, it estimates the next dominant symbol from the residue signal. Unlike CLEAN-BLAST, however, which at this time estimates the third strongest symbol, RELAX-BLAST instead updates the two already detected symbols in an iterative manner until the difference of the RELAX-BLAST estimates between two successive iterations becomes less than a certain threshold. Once these two symbols are subtracted from the measurements and the third strongest symbol is estimated, the three symbols are again updated in an iterative manner until all the three estimates do not improve anymore. This process is repeated until all the N symbols are detected and updated. Finally, note that when $N = 1$, i.e., for a SIMO or SISO system, LCN, CLEAN-BLAST and RELAX-BLAST become identical approaches.

2.4 Numerical and Experimental Results

In this section we evaluate the performance of the CA training sequences, compare IAA with MP, OMP and LSMP for channel estimation and compare CLEAN-BLAST with RELAX-BLAST for symbol detection using simulations and/or the RACE08 experimental results. Throughout this section, all the CIR estimation algorithms are followed by BIC to achieve sparsity.

2.4.1 Simulations

2.4.1.1 Channel estimation performance

To begin with, we consider the problem of CIR estimation for a 4×1 multi-input single-output (MISO) system with a time-invariant channel. The simulated CIR

coefficients resemble real UWA conditions encountered in the RACE08 experimental measurements. Figure 2-3 shows the modulus of the CIRs corresponding to the four transmitters where $R = 30$ delay taps are considered. Given the training symbols, the received data samples are constructed using (2–5), where $\mathbf{e}_1$ is assumed to be a circularly symmetric independent and identically distributed (i.i.d.) complex-valued Gaussian random process with mean zero and variance σ^2.

Figure 2-4 shows the mean squared error (MSE) of the channel estimates obtained by MP, OMP, LSMP and IAA with two different training sequences: QPSK training and CA training. In QPSK training, each training symbol is randomly selected to be one of the four QPSK constellation points whereas in CA training each symbol is selected by using the CA algorithm described in Section 2.2.2. The training sequence length is set at $P = 128$ symbols. Each point in Figure 2-4 is obtained by averaging 100 Monte-Carlo trials. We observe that when the QPSK training is used, IAA significantly outperforms the other channel estimation methods. OMP and LSMP show similar performance whereas MP shows the worst performance. On the other hand, when the CA training sequences are used, the performance gap between IAA and the MP based channel estimation algorithms diminishes and all algorithms yield very similar performance although IAA still gives the lowest MSE. Moreover, the performance of IAA is not affected very much from the characteristics of the training sequences used. This is an advantage over the other methods since in the decision-directed mode, the channel has to be updated using the previously decoded symbols, which do not have as good auto- and cross-correlation properties as the specifically designed training sequences.

2.4.1.2 Symbol detection performance

We now evaluate the bit error rates (BER) of CLEAN-BLAST and RELAX-BLAST for a 4×12 MIMO system. The package structure shown in Figure 2-1 is used in the simulations with CA training sequences consisting of $P = 512$ symbols, a payload sequence consisting of $Q = 6000$ QPSK modulated symbols and two gaps consisting of

80 mute symbols each. IAA is used for channel estimation. The detection order for the algorithms is 3, 2, 4, 1 (i.e., the third channel is assumed to have the strongest channel response at all the receivers and the first channel the weakest). The average BERs over 100 Monte-Carlo trials are shown for the data transmitted from all four transducers in Figure 2-5. We observe that RELAX-BLAST shows much better performance than CLEAN-BLAST as long as severe error propagation does not exist. This result is supported by the fact that similar performance improvements in spectral estimation are obtained when RELAX is used instead of CLEAN [38, 39]. Due to this reason, we will use RELAX-BLAST when analyzing the RACE08 data in the following.

2.4.2 RACE08 In-Water Experimentation Results

In this part, we evaluate our proposed MIMO underwater communications scheme using the RACE08 experimental data set. RACE08 was conducted by WHOI in Narragansett Bay. The water depths ranged from 9 to 14 m during the experiments. Surface conditions were primarily wind blown chop. A 4×24 MIMO system was used in the experiments. The primary transmitter was located approximately 4 m above the bottom of the ocean using a stationary tripod. Below the primary transmitter, a source array consisting of 3 transducers was deployed vertically with a spacing of 0.6 m between the elements. The top element of the source array was 1 m below the primary source. 24 receiving transducers were mounted at a range of approximately 400 m. Receivers were deployed vertically with a spacing of 0.05 m between the individual elements. The carrier frequency and the bandwidth employed in the RACE08 experiments were 12 kHz and 3.9 kHz, respectively.

The data packet that we will consider herein is from epoch “0830156”. Some epochs could not be evaluated due to the severe conditions of sea. Among the many epochs that result in reasonable performance, epoch “0830156” was chosen arbitrarily. The package structure shown in Figure 2-1 was used in the experiments with CA training sequences consisting of $P = 512$ symbols, a payload sequence consisting

of $Q = 2000$ QPSK modulated symbols and two gaps consisting of 80 mute symbols each. The symbol rate was 3906.25 symbols per second. By applying QPSK modulation and using 4 transmitters simultaneously, a 31.25 kbps uncoded payload data rate was achieved. The coding scheme we used for the experiments was a 1/2 convolutional encoder with constraint length of 5, and generator polynomials (1 0 0 1 1) and (1 1 0 1 1) [57]. This coding scheme reduces the net payload bit rate to 15.63 kbps.

The selection of the number of delay taps, R, to consider is very important. A value too small will loose important channel features whereas a value too large will complicate the receiver and may result in overfitting as well as increased noise. We found out empirically that $R = 30$ yields reasonable results. Figure 2-6 shows the modulus of the training-directed IAA estimate of the CIR at receiver 1. The CIRs for the other receivers, i.e., $\{\hat{\mathbf{h}}_m\}_{m=2}^{24}$ share similar structure with $\hat{\mathbf{h}}_1$. As shown in Figure 2-6, the detection order should be 2 (strongest coefficients), 4, 3 and 1 (weakest coefficients).

The channel tracking approach we follow is summarized in Figure 2-7. In the first step, the CIR is estimated using the training sequences. Based on the initial CIR estimate, the first $L + 50$ payload symbols are obtained using RELAX-BLAST, where L was defined after (2–7). Next, a decision-directed CIR estimation is done using the first L estimated symbols. The reason for not using all the $L + 50$ estimated symbols will be explained shortly. With the updated CIR, starting from the $(L - 49)^{th}$ symbol, the subsequent $L + 100$ symbols are detected again using RELAX-BLAST. This process is repeated until all the 2000 payload symbols are detected. Figure 2-7 shows that 100 more symbols (50 more symbols at the first and last steps) are detected other than the L symbols used to update the CIR at each step. These 50 margin symbols on either end serve as guard intervals because the errors tend to happen at the beginning and end of each block. This is partly due to no mute symbols being available within the payload sequence.

Table 2-4. Bit error rate (BER) for $L = 200$.

	Uncoded BER (%)				Coded BER (%)			
	Tx 1	Tx 2	Tx 3	Tx 4	Tx 1	Tx 2	Tx 3	Tx 4
MP	30.45	6.80	14.38	3.83	46.70	0	8.85	0
OMP	12.15	0.60	2.00	0.35	2.40	0	0	0
LSMP	12.15	0.60	2.00	0.35	2.40	0	0	0
IAA	4.63	0.10	0.35	0	0	0	0	0

In Table 2-4 we show the uncoded and coded BERs obtained via MP, OMP, LSMP or IAA as the channel estimation algorithm. For the results presented in this table, the number of payload symbols used for updating the channel coefficients is 200, i.e., $L = 200$. We observe that IAA provides the best performance among all four algorithms. The average uncoded BER for IAA is 1.27%, MP is 13.86%, and OMP and LSMP is 3.78% and the coded average BER for IAA is 0%, MP is 13.89%, and OMP and LSMP is 0.6%. As expected, the sequence with the strongest (weakest) channel coefficients is estimated with the highest (lowest) accuracy; see Figure 2-6.

In Table 2-5 the uncoded and coded BERs are shown for $L = 400$. This means that the channel will be updated less frequently than in the case where $L = 200$. We observe that now IAA, OMP and LSMP show almost identical performance. The average uncoded BER for IAA is 0.38%, MP is 2.09%, and OMP and LSMP is 0.37% and the coded average BER for IAA is 0%, MP is 0.01%, and OMP and LSMP is 0%. As we mentioned previously, when L is large or the sequence used for updating the channel is well-structured, the performance of MP type of algorithms approaches that of IAA. However, it might not be always possible to select L large in practice.

The choice of L determines the rate at which the CIR will be updated in the decision-directed mode. It also determines the accuracy of the CIRs. As previously mentioned, the larger the L, the more accurate the channel estimates will be assuming that the previously detected symbols are correct and the channel is stationary. On

Table 2-5. BER for $L = 400$.

	Uncoded BER (%)				Coded BER (%)			
	Tx 1	Tx 2	Tx 3	Tx 4	Tx 1	Tx 2	Tx 3	Tx 4
MP	6.98	0.23	1.05	0.13	0.05	0	0	0
OMP	1.48	0	0	0	0	0	0	0
LSMP	1.48	0	0	0	0	0	0	0
IAA	1.50	0	0	0	0	0	0	0

the other hand, for larger L, the channel will be updated less frequently and hence the results will be inaccurate for a rapidly varying channel. Therefore, the choice of L has a direct effect on the performances of MP, OMP, LSMP and IAA. Moreover, L also determines the computational complexities of these algorithms. For the current set of data, we observed that the channel is rather benign and using a large L value results in better estimates than using a lower one as seen in Tables 2-4 and 2-5. However, for a rapidly varying channel where L has to be selected small, IAA appears to be the best candidate for channel estimation as its performance is still good with small L whereas MP type of algorithms show relatively worse performance. Note that in our experiments, neither the training sequence length P nor the gap lengths have been optimized for the best performance as no prior information of the experimental conditions was available. Moreover, for the current experimental conditions, a 1/2 rate convolutional code appears to be on the conservative side to achieve zero coded BER.

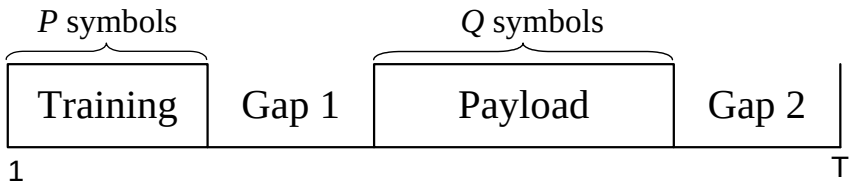


Figure 2-1. The structure of a single data package.

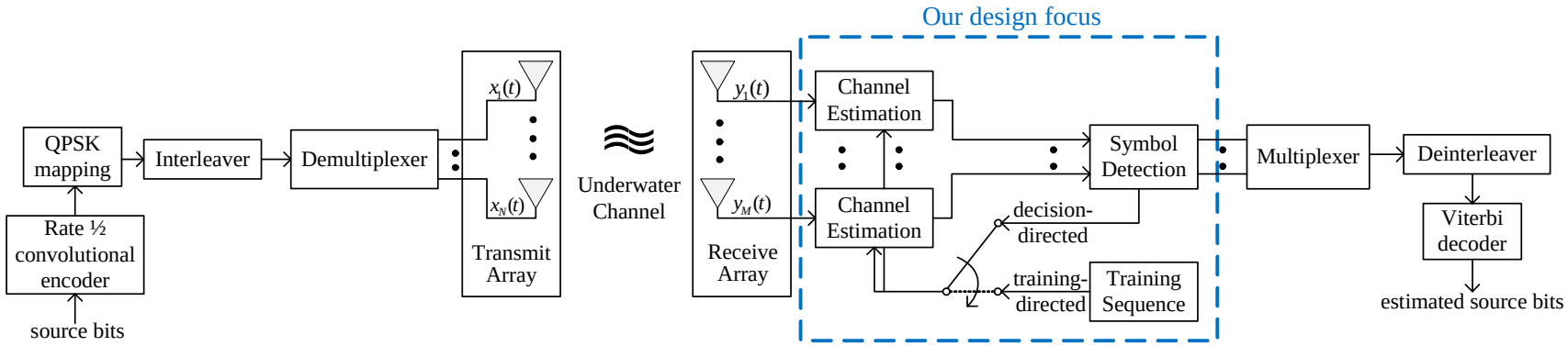


Figure 2-2. An $N \times M$ MIMO UAC system. The blocks inside the dashed rectangle are the focus of our attention in this chapter.

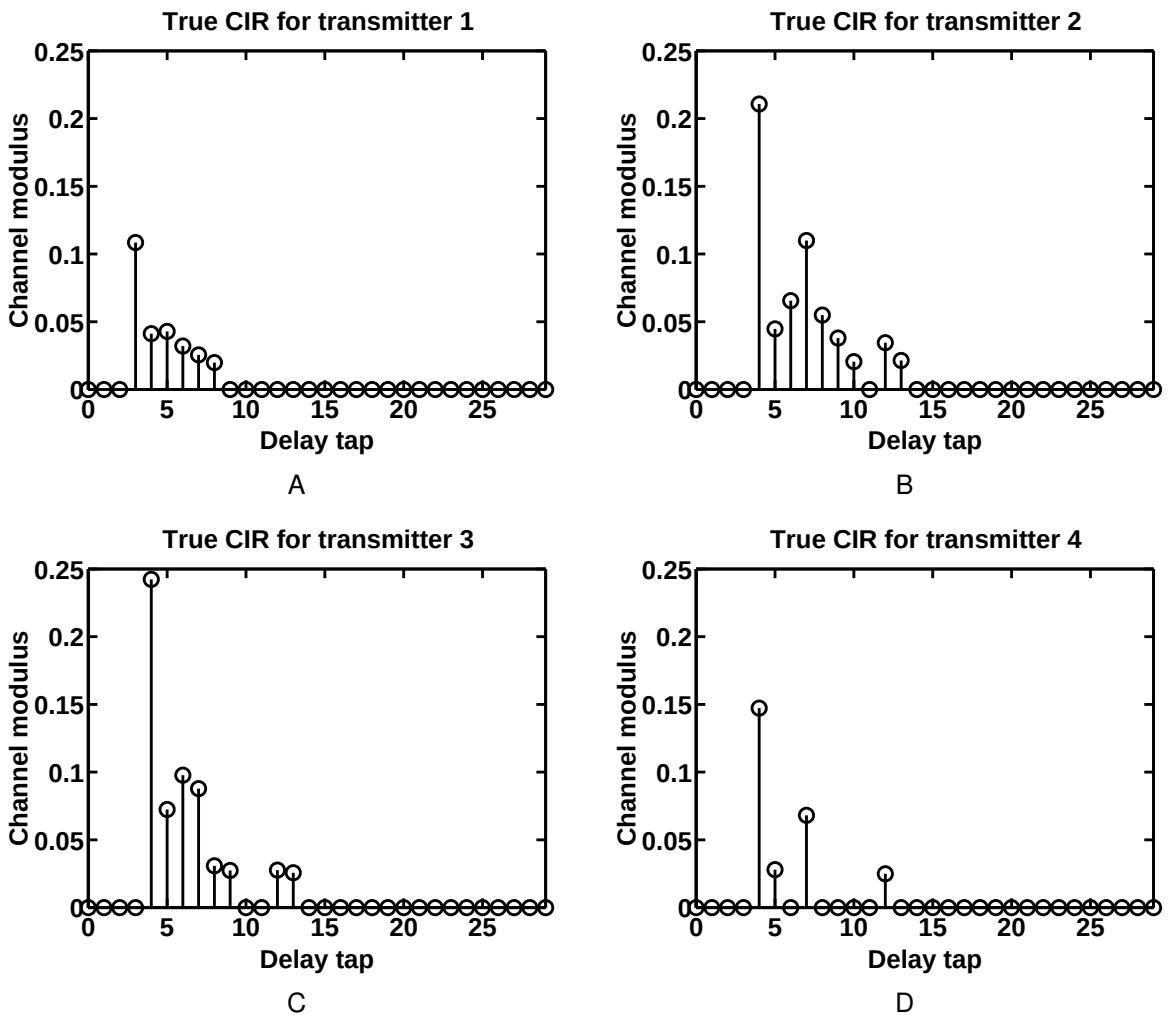


Figure 2-3. The modulus of the simulated CIRs between the four transmitters and the receiver in a 4×1 MISO system.

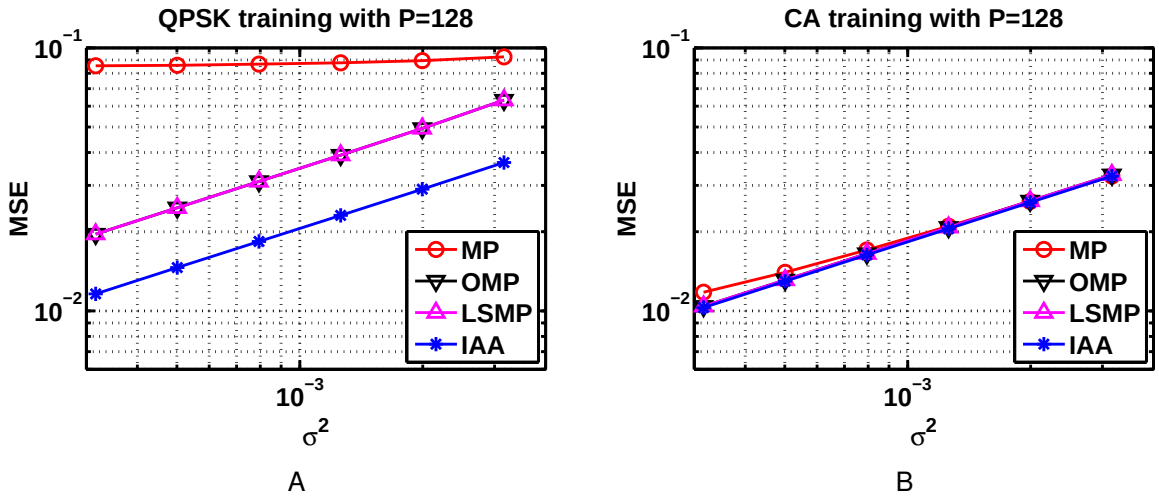


Figure 2-4. MSE of the CIR estimates for a 4×1 MISO system using the QPSK and CA training sequences with $P = 128$ symbols. Each point is averaged over 100 Monte-Carlo trials.

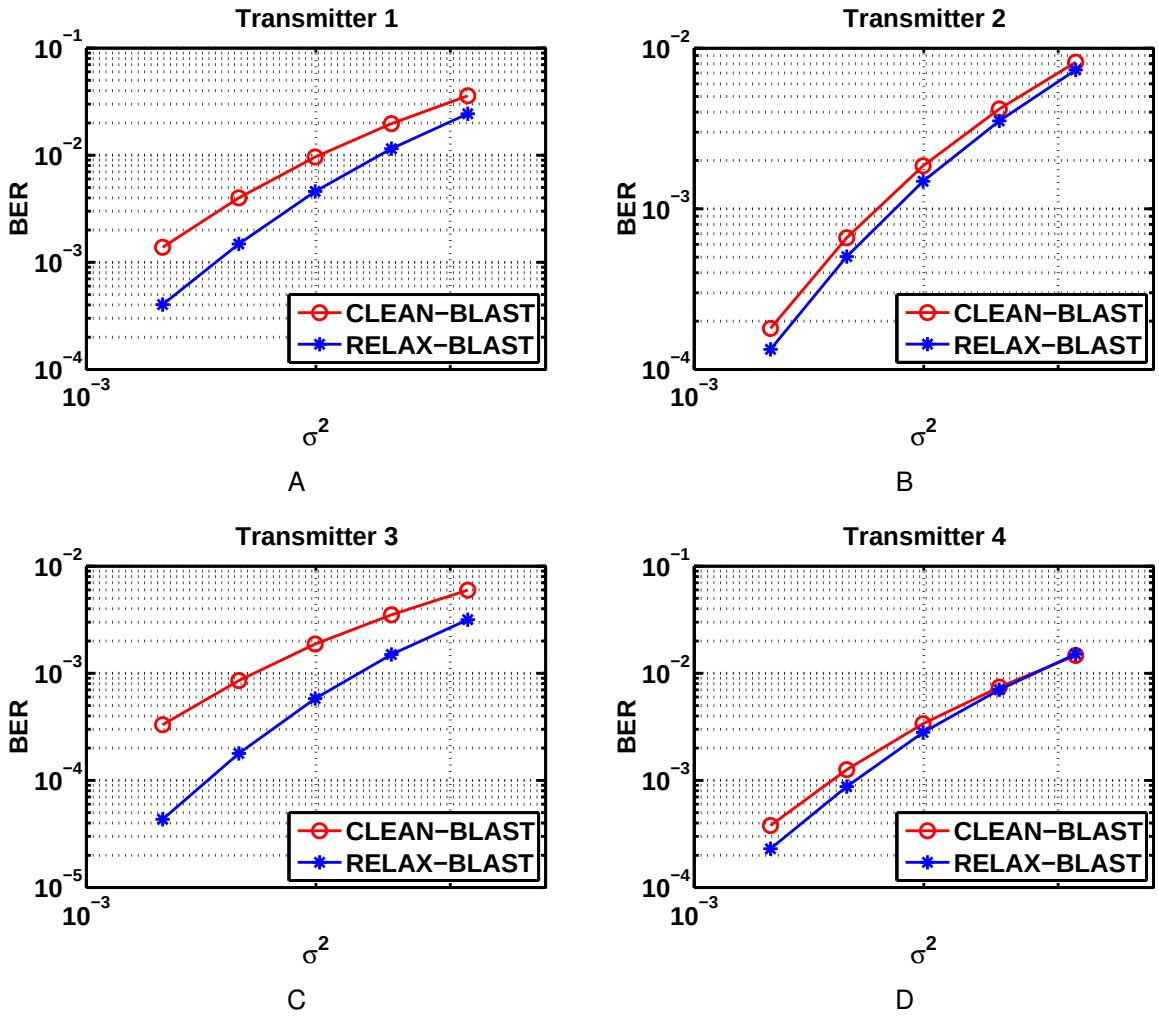


Figure 2-5. The BERs of each of the four transmitted payload sequences for a 4×12 MIMO system. The training sequences consist of $P = 512$ symbols and are designed by the CA algorithm. The detection performance of CLEAN-BLAST and RELAX-BLAST are compared in terms of BER averaged over 100 independent Monte-Carlo trials for varying levels of the noise variance σ^2.

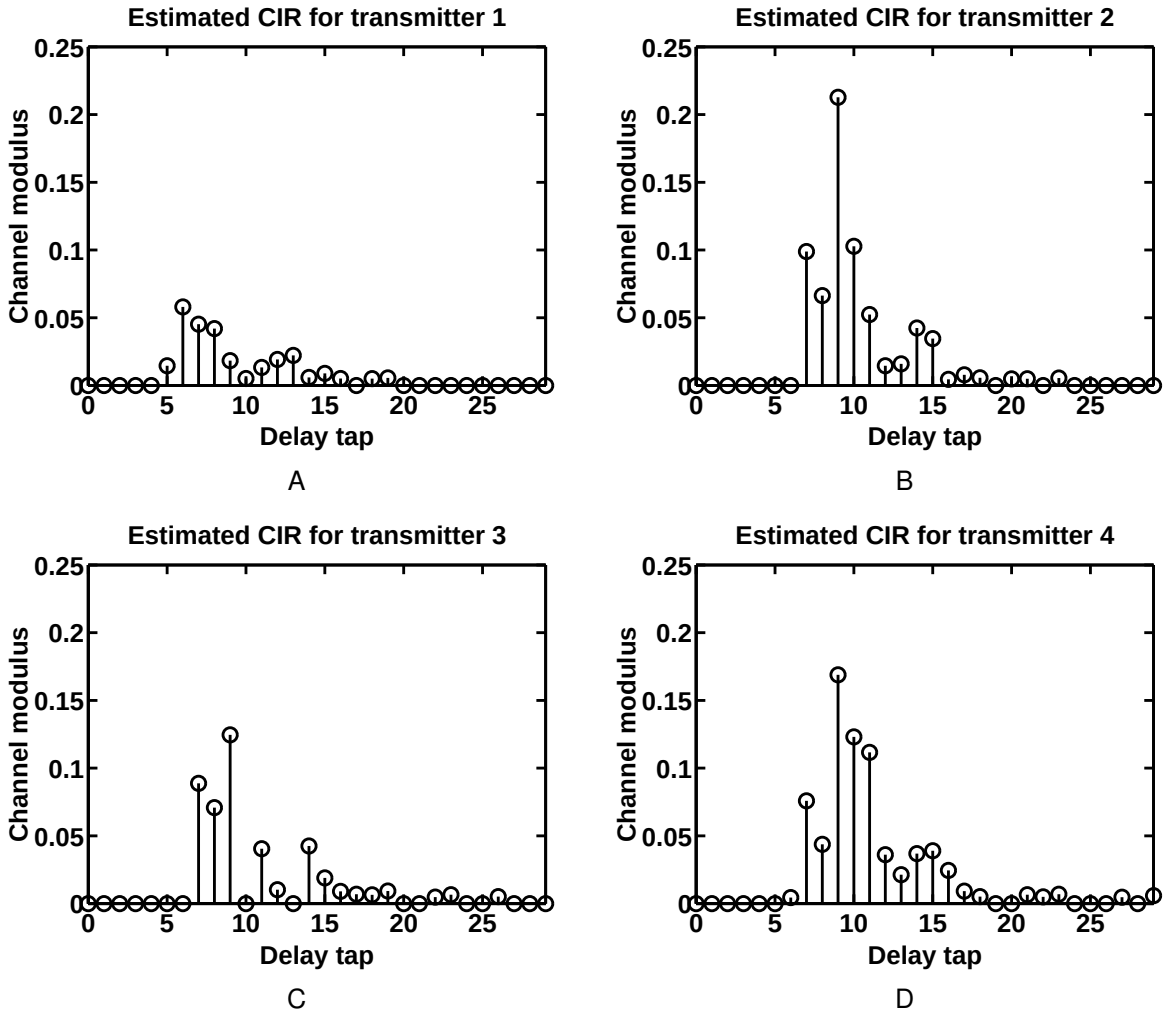


Figure 2-6. The modulus of the four RACE08 CIRs estimated by IAA for the first receiver from epoch “0830156”.

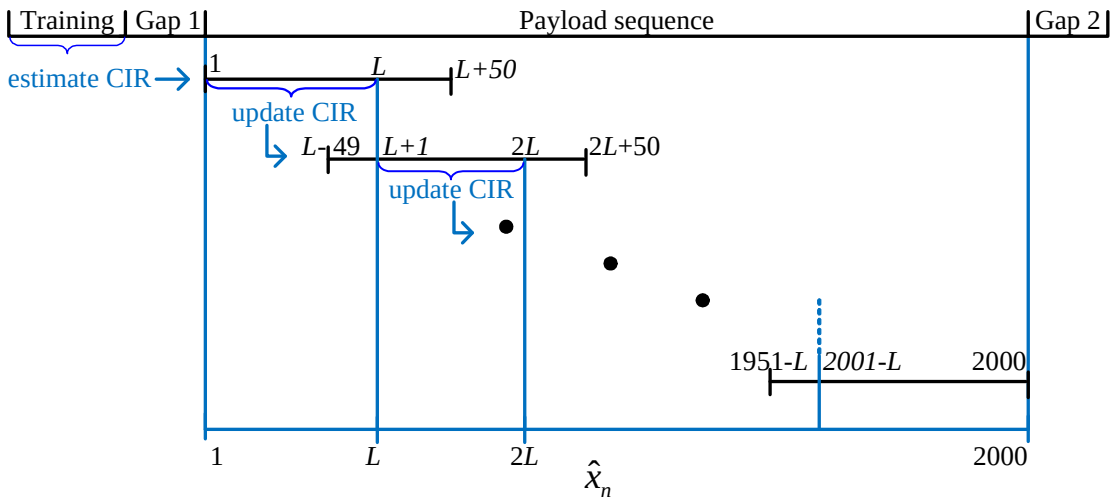


Figure 2-7. The channel tracking procedure.

CHAPTER 3
ON BAYESIAN CHANNEL ESTIMATION AND FAST FOURIER TRANSFORM BASED SYMBOL DETECTION IN MIMO UAC

In this chapter, several new MIMO UAC schemes are developed to improve the overall efficiency and performance of UAC systems. We still address the two important aspects of coded MIMO UAC systems: *i)* enhanced estimation of the underwater CIR and *ii)* efficient symbol detection in the presence of severe interference. More specifically, we propose a Bayesian channel estimation algorithm that provides good channel estimation performance along with reduced computational complexity compared to IAA. Moreover, in order to pursue real-time implementation, we exploit the conjugate gradient method and diagonalization properties of the circulant matrix to significantly speed up symbol detection. The proposed MIMO UAC techniques are thoroughly tested using in-water experimental measurements recently acquired by WHOI during the 2008 Surface Processes and Acoustic Communications Experiment (SPACE08).

Aside from the fact that IAA is user parameter free, it was shown in Chapter 2 that it outperforms MP based algorithms. Meanwhile, the computational complexity required by IAA is somewhat high. In this chapter, a novel, user parameter free method is presented as an alternative to the aforementioned methods for sparse channel estimation. This is a maximum a posteriori (MAP) based Bayesian approach, referred to as sparse learning via iterative minimization (SLIM), As shown later, the channel estimation performance of SLIM is similar to that of IAA but at a considerably lower computational cost. In addition, SLIM generates not only CIR estimates, but also the variance thereof quantifying the confidence in the SLIM estimate. By making use of these first- and second-order statistics, a scheme to automatically determine the number of relevant channel taps is also presented.

Another important aspect of digital communications is symbol detection given the estimated channel coefficients. It was demonstrated in Chapter 2 that RELAX-BLAST provides better performance than LCN and V-BLAST. Although RELAX-BLAST does

not need the computation of as many filters as MIMO decision feedback equalizer (MIMO-DFE) [59], it is still relatively more complex compared to V-BLAST. This issue is addressed herein by the efficient implementation of RELAX-BLAST using the conjugate gradient (CG) method [24, 62]. Moreover, the speed of the CG method is significantly improved by using a fast Fourier transform (FFT)-based matrix decomposition technique that exploits the circulant properties of the channel matrix [18, 77].

The rest of this chapter is organized as follows. Section 3.1 outlines the system configuration and describes the considered data package structure. Section 3.2 formulates the CIR estimation problem and presents the SLIM algorithm for channel estimation together with a scheme of automatically selecting the channel length. Next, an overview of the symbol detection problem is provided and a CG based method to improve the efficiency of RELAX-BLAST is presented in Section 3.3. Section 3.4 presents simulation results, as well as in-water experimental results using the data gathered in the SPACE08 experiment.

3.1 System Outline

Consider an $N \times M$ MIMO UAC system equipped with N transmit transducers and M receive hydrophones. The data stream of each transmitter forms a package and each package consists of a sequence of packets. Each packet comprises a training sequence followed by a payload sequence, as shown in Figure 3-1. It is worth pointing out that for the sake of increasing the data rate and preventing the CIR estimates from outdating, the guard intervals between the training sequence and payload sequence, as designed for the RACE08 experiment (Figure 2-1), are not present here. The payload sequence contains the data to be transmitted and is generally encoded by some coding scheme. Rather than encoding the entire payload sequence at once, the sequence is divided into several blocks, each of which is encoded separately. This coding scheme improves the accuracy of the payload symbol estimates and consequently results in better overall performance by mitigating error propagation. For each payload data block, convolutional

encoding and random interleaving are employed. Also, separate encoders are used for the different transmitters. Furthermore, Gray coded quadrature phase-shift keying (QPSK) [57] is used to map bits into symbols. The four constellation points of QPSK symbols, i.e., $\{e^{j(2n-1)\frac{\pi}{4}}\}_{n=1}^{4}$, lie on the unit circle and have unit modulus, a property that is desirable from an amplifier efficiency point of view. Similarly, the training symbols should also have unit modulus, while no restriction is imposed on their phase values.

As shown in Figure 3-2, the measured signals are first passed through an equalizer, which consists of two steps: 1) CIR estimation (in training- or decision-directed mode) and 2) symbol detection. By reversing the steps in the symbol generation process, the symbol estimates at the equalizer output are then demapped, deinterleaved and finally decoded using a Max-Log-MAP approach [58]. In addition, the estimated source bits are used to enhance the equalizer performance; this feedback mechanism is referred to as interference cancellation [84]. In what follows, our consideration is confined to one data packet of the form given in Figure 3-1 (the same analysis is repeated for all data packets of interest). It is assumed that the sampling and synchronization procedures have already been employed, and that the sampled complex baseband signals are available at the receiver.

3.2 Channel Estimation

3.2.1 Training-Directed Mode

The training-directed channel estimation problem is almost the same as that presented in Section 2.2.1.1. In the absence of the guard interval between the training sequence and payload sequence (Figure 3-1), $\tilde{\mathbf{X}}_n$ is constructed as:

$$\tilde{\mathbf{X}}_n = \begin{bmatrix} x_n(1) & 0 & \dots & 0 \\ x_n(2) & x_n(1) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ x_n(P) & x_n(P-1) & \dots & x_n(P-R+1) \end{bmatrix}, \quad n = 1, \dots, N. \tag{3–1}$$

By comparing (3–1) with (2–4), one can see that $\tilde{\mathbf{X}}_n$ in (3–1) can be obtained by discarding the bottom $R-1$ rows from $\tilde{\mathbf{X}}_n$ in (2–4). To conform with the dimensions of $\tilde{\mathbf{X}}_n$ in (3–1), The measurement vector at the mth receiver is given by:

$$\mathbf{y}_m = [y_m(1), \dots, y_m(P)]^T, \quad m = 1, \dots, M, \tag{3–2}$$

which contains the P synchronized measured symbols (for instance, $\{y_m(1)\}$ maps to $\{x_n(1)\}$) at the mth receiver. With the new definitions for $\mathbf{y}_m$ and $\tilde{\mathbf{X}}_n$, (2–5) can still be used. Training-directed channel estimation problelm, once again, reduces to estimating the channels $\mathbf{h}_m$ from the measurements $\mathbf{y}_m$ and known $\mathbf{X}$.

3.2.2 Channel Estimation Algorithm

The channel estimation problem at each receiver, in either training-directed mode (Equation (2–5)) or decision-directed mode (Section 2.2.1.2 or Section 3.2.4), has the generic form given by

$$\mathbf{y} = \mathbf{X}\mathbf{h} + \mathbf{e}. \tag{3–3}$$

The receiver index m is omitted here for notational simplicity. We note that the channel estimation at each receiver can be done in parallel, and we also remark that the number of elements in $\mathbf{y}$, namely d_y, might vary in different modes. $\mathbf{e}$ in (3–3) is assumed to contain circularly symmetric independent and identically distributed (i.i.d.) complex-valued Gaussian random variables with zero mean and variance η, denoted as $\mathbf{e} \sim \mathcal{CN}(\mathbf{0}, \eta\mathbf{I})$ (The practical validity of this assumption will be verified by analyzing experimental ambient noise in Section 3.4.2.2). The problem is then to estimate $\mathbf{h}$, given $\mathbf{y}$ and $\mathbf{X}$. In UAC systems, the channel $\mathbf{h}$ is usually sparse, i.e., although it contains NR unknowns, many of these can be approximated as zero. We present the SLIM algorithm to solve this sparse channel estimation problem. Note that since $\mathbf{h}$ contains the CIR of all N transmitters, the SLIM algorithm will estimate them simultaneously.

Consider the following hierarchical Bayesian model:

$$\mathbf{y}|\mathbf{h}, \eta \sim \mathcal{CN}(\mathbf{X}\mathbf{h}, \eta\mathbf{I}), \tag{3–4}$$

$$\mathbf{h}|\mathbf{p} \sim \mathcal{CN}(\mathbf{0}, \mathbf{P}), \tag{3–5}$$

where (3–4) follows directly from the assumption $\mathbf{e} \sim \mathcal{CN}(\mathbf{0}, \eta\mathbf{I})$. By assuming independent channel taps, the covariance matrix $\mathbf{P}$ in (3–5) becomes diagonal $\mathbf{P} = \mathrm{diag}(\mathbf{p})$ with $\mathbf{p} = [p_1, p_2, \dots, p_{NR}]^T$, where p_n is the variance of h_n, the nth element of $\mathbf{h}$. Furthermore, by considering a flat prior on both η and $\{p_n\}_{n=1}^{NR}$, the channel vector $\mathbf{h}$, the covariance matrix $\mathbf{P}$ (or more precisely, its diagonal elements $\mathbf{p}$) and the noise power η can be estimated based on the maximum a posteriori (MAP) criterion:

$$\max_{\mathbf{h},\mathbf{p},\eta} p(\mathbf{h}, \mathbf{p}, \eta|\mathbf{y}) = \max_{\mathbf{h},\mathbf{p},\eta} p(\mathbf{y}|\mathbf{h}, \eta)p(\mathbf{h}|\mathbf{p}). \tag{3–6}$$

By combining (3–4), (3–5) and (3–6), and by taking the negative logarithm of the cost function, the optimization problem formulated in (3–6) becomes

$$\min_{\mathbf{h},\mathbf{p},\eta} \left(d_y \log \eta + \frac{\|\mathbf{y} - \mathbf{X}\mathbf{h}\|^2}{\eta} + \sum_{n=1}^{NR} \log p_n + \sum_{n=1}^{NR} \frac{|h_n|^2}{p_n} \right), \tag{3–7}$$

which can be solved using an alternating approach (also known as the coordinate descent method [85]): at each iteration, one of the parameters $\mathbf{h}$, $\mathbf{p}$ and η is updated while keeping the other two fixed. In this way, the single difficult optimization problem is divided into 3 simpler subproblems. SLIM keeps iterating until a predefined number of iterations is reached. Under mild conditions, the cyclic optimization scheme guarantees that the SLIM algorithm converges, at least to a local minimum of (3–7) [101].

The 4 steps of the SLIM algorithm at the tth iteration are outlined below:

1. Given $\mathbf{h}^{(t-1)}$, the optimal $\mathbf{P}^{(t)}$ that minimizes the cost function in (3–7) is given by:

$$p_n^{(t)} = \left|h_n^{(t-1)}\right|^2 \qquad n = 1, \dots, NR. \tag{3–8}$$

This result can be found by taking the partial derivative of (3–7) with respect to p_n (with all the terms irrelevant to p_n dropped):

$$\frac{\partial}{\partial p_n}\left(\log p_n + \frac{\left|h_n^{(t-1)}\right|^2}{p_n}\right) = \frac{1}{p_n} - \frac{\left|h_n^{(t-1)}\right|^2}{p_n^2}. \tag{3–9}$$

Setting the above partial derivative to 0 yields the optimal value $p_n^{(t)}$ in (3–8). For better numerical stability, we set $p_n^{(t)}$ (or equivalently $h_n^{(t)}$) to zero if $p_n^{(t)} < 10^{-15}$. Note that the SLIM algorithm achieves sparsity due to the hierarchical Bayesian model, not through this "compare-and-null" step.

2. Once $\mathbf{P}^{(t)}$ is available, we proceed to update the CIR estimate. Equivalently, $\mathbf{h}^{(t)}$ is obtained by taking the partial derivative of (3–7) with respect to $\mathbf{h}$ and setting the result to zero. By noticing that $\sum_{n=1}^{NR} \frac{|h_n|^2}{p_n^{(t)}} = \mathbf{h}^H \left(\mathbf{P}^{(t)}\right)^{-1} \mathbf{h}$ and by solving

$$\frac{\partial}{\partial \mathbf{h}}\left(\frac{\|\mathbf{y}-\mathbf{X}\mathbf{h}\|^2}{\eta^{(t-1)}} + \mathbf{h}^H \left(\mathbf{P}^{(t)}\right)^{-1} \mathbf{h}\right) = \left[\frac{\mathbf{X}^H\mathbf{X}}{\eta^{(t-1)}} + \left(\mathbf{P}^{(t)}\right)^{-1}\right]\mathbf{h} - \frac{\mathbf{X}^H\mathbf{y}}{\eta^{(t-1)}} = 0, \tag{3–10}$$

we get:

$$\mathbf{h}^{(t)} = \left[\mathbf{X}^H\mathbf{X} + \eta^{(t-1)}\left(\mathbf{P}^{(t)}\right)^{-1}\right]^{-1}\mathbf{X}^H\mathbf{y}. \tag{3–11}$$

While inverting $\mathbf{P}^{(t)}$, its zero diagonal entries are removed, and the associated columns in $\mathbf{X}$ are discarded.

3. Using the most recently obtained $\mathbf{h}^{(t)}$ in (3–11), we finally estimate the noise power by solving:

$$\frac{\partial}{\partial \eta}\left(d_y \log \eta + \frac{\left\|\mathbf{y}-\mathbf{X}\mathbf{h}^{(t)}\right\|^2}{\eta}\right) = \frac{d_y}{\eta} - \frac{\left\|\mathbf{y}-\mathbf{X}\mathbf{h}^{(t)}\right\|^2}{\eta^2} = 0. \tag{3–12}$$

The solution is given by:

$$\eta^{(t)} = \frac{1}{d_y}\left\|\mathbf{y}-\mathbf{X}\mathbf{h}^{(t)}\right\|^2. \tag{3–13}$$

4. $t = t+1$. Go back to Step 1 if t is less than the predefined iteration number, or terminate otherwise.

In training-directed mode, the channel characteristics are in general not available a priori. In our examples, $\mathbf{h}^{(0)}$ is initialized using the standard matched filter, and the noise power $\eta^{(0)}$ is initialized with a small positive number, for instance, 10^{-10}. Our empirical experience suggests that the SLIM algorithm does not provide significant performance

improvements after about 15 iterations. Next, we proceed to compare the computational complexity between SLIM and IAA. As presented in [48], each IAA iteration has a complexity on the scale of $\mathcal{O}(d_y^3)$, where d_y is defined after (3–3). Referring to the 3 update tasks involved at each SLIM iteration, we can see that the bottleneck of SLIM stems from the matrix inverse in (3–11), which has a complexity $\mathcal{O}(N^3R^3)$. Therefore, the complexity of SLIM is smaller than that of IAA when $NR < d_y$ (after automatic selection of channel tap number presented in Section III-C, the complexity of each SLIM iteration can be further reduced to $\mathcal{O}(N^3\tilde{R}^3)$ with $\tilde{R} \leq R$, and the inequality $N\tilde{R} < d_y$ is satisfied in most practical cases, which verifies the efficiency of SLIM over IAA).

3.2.3 Automatic Selection of Channel Tap Number

In practical UAC, automatic determination of the number of channel taps, denoted as $\tilde{R}$, is an important task. By setting a small value, important channel features might be lost, whereas a large value will result in overfitting as well as increased noise levels. In the absence of channel truth, the tuning of $\tilde{R}$ often relies on trial-and-error and involves user intervention. We avoid this problem by making use of the SLIM estimates as follows. The hierarchical Bayesian model presented in (3–4) and (3–5) leads to [79]:

$$\mathbf{h}|\mathbf{y}, \mathbf{p}^*, \eta^* \sim \mathcal{CN}(\mu, \mathbf{\Sigma}), \tag{3–14}$$

where $\mathbf{p}^*$ and η^* denote the solutions to the optimization problem in (3–7). The mean vector μ and the covariance matrix $\mathbf{\Sigma}$ can be expressed, respectively, as

$$\mu = \left(\mathbf{X}^H\mathbf{X} + \eta^*\mathbf{P}^{*-1}\right)^{-1}\mathbf{X}^H\mathbf{y}, \tag{3–15}$$

and

$$\mathbf{\Sigma} = \mathbf{P}^* - \mathbf{P}^*\mathbf{X}^H\left(\mathbf{X}\mathbf{P}^*\mathbf{X}^H + \eta^*\mathbf{I}\right)^{-1}\mathbf{X}\mathbf{P}^*. \tag{3–16}$$

The posterior density $\mathbf{h}$ given $\mathbf{y}$, i.e., $p(\mathbf{h}|\mathbf{y})$, can be obtained by integrating out the variables $\mathbf{p}$ and η in $p(\mathbf{h}|\mathbf{y}, \mathbf{p}, \eta)$:

$$\begin{aligned} p(\mathbf{h}|\mathbf{y}) &= \int\int p(\mathbf{h}|\mathbf{y}, \mathbf{p}, \eta) p(\mathbf{p}) p(\eta) d\mathbf{p} d\eta \\ &\approx p(\hat{\mathbf{h}}|\mathbf{y}, \mathbf{P}^*, \eta^*). \end{aligned} \tag{3–17}$$

The approximation above follows by assuming that both $p(\mathbf{p})$ and $p(\eta)$ can be modeled as Dirac functions [79], i.e., $p(\mathbf{p}) = \prod_{n=1}^{NR} \delta(p_n - p_n^*)$ and $p(\eta) = \delta(\eta - \eta^*)$. Consequently, we can approximately consider

$$\mathbf{h}|\mathbf{y} \sim \mathcal{CN}(\mu, \mathbf{\Sigma}). \tag{3–18}$$

So far, the discussions and derivations are developed with the receiver index m omitted. The subsequent derivations, however, require incorporating the receiver index. For clarity, we hence rewrite (3–18) as

$$\mathbf{h}_m|\mathbf{y}_m \sim \mathcal{CN}(\mu_m, \mathbf{\Sigma}_m), \qquad m = 1, \dots, M, \tag{3–19}$$

to remind us that the results are with respect to the mth receiver. Under the assumption that the channel taps are independent, each element of $\mathbf{h}_m$ given $\mathbf{y}_m$ can individually be characterized through a complex-valued Gaussian statistic. Using (2–3) and the definition of $\mathbf{h}_m$ after (2–5), we have

$$h_{n,m}(r)|\mathbf{y}_m \sim \mathcal{CN}(\mu_{n,m}(r), \sigma_{n,m}^2(r)), \tag{3–20}$$

where $n = 1, \dots, N$, $m = 1, \dots, M$ and $r = 1, \dots, R$. The mean $\mu_{n,m}(r)$ and the variance $\sigma_{n,m}^2(r)$ are appropriate entries of μ_m and diagonal entry of $\mathbf{\Sigma}_m$, respectively.

Let

$$\beta_{n,m}(r) = \frac{|\mu_{n,m}(r)|}{\sigma_{n,m}(r)} \quad \text{and} \quad \bar{\beta}_n(r) = \frac{1}{M}\sum_{m=1}^{M} \beta_{n,m}(r). \tag{3–21}$$

Intuitively, the real-valued quantity $\beta_{n,m}(r)$ in (3–21) quantifies our confidence in keeping the rth tap as an effective channel tap for the nth transmitter, which is more likely to

happen when the channel modulus is large (i.e., $|\mu_{n,m}(r)|$ is large) and we are confident of the estimate (i.e., $\sigma_{n,m}(r)$ is small). The proposed channel tap selection scheme is then to use $\bar{\beta}_n(r)$ as follows: for the nth transmitter, we start by testing the last (i.e., the Rth) channel tap. If $\bar{\beta}_n(R) < 3$ (i.e., 3 times the standard derivation, note that $\beta_{n,m}(R)$ is a normalized statistic), we proceed to test the $(R-1)^{\text{st}}$ channel tap and continue until we come across a channel tap, say the $\tilde{R}_n^{\text{th}}$ tap, that renders $\bar{\beta}_n(\tilde{R}_n) \geq 3$ for the first time. This procedure is then repeated for $n = 1, \ldots, N$, i.e., N times, and in such a way that we obtain $\tilde{R}_n$ for each transmitter. Finally, $\tilde{R}$ is determined as $\tilde{R} = \min\{\tilde{R}_n\}_{n=1}^N$ (using $\tilde{R} = \max\{\tilde{R}_n\}_{n=1}^N$ yields similar detection performance). The so-obtained $\tilde{R}$ is then used for subsequent processing.

3.2.4 Decision-Directed Mode

The problem in decision-directed mode is almost identical to that presented in Section 2.2.1.2 except that R is now replaced by $\tilde{R}$, the automatically determined channel length (Section 3.2.3). That is, $\tilde{\mathbf{X}}_n$ is constructed as:

$$\tilde{\mathbf{X}}_n = \begin{bmatrix} \hat{x}_n(t_i) & \hat{x}_n(t_i-1) & \ldots & \hat{x}_n(t_i-\tilde{R}+1) \\ \hat{x}_n(t_i+1) & \hat{x}_n(t_i) & \ldots & \hat{x}_n(t_i-\tilde{R}+2) \\ \vdots & \vdots & & \vdots \\ \hat{x}_n(t_f) & \hat{x}_n(t_f-1) & \ldots & \hat{x}_n(t_f-\tilde{R}+1) \end{bmatrix}, \quad n = 1, \ldots, N. \tag{3–22}$$

When conducting decision-directed channel tracking, for the sake of computational efficiency, $\mathbf{p}^{(0)}$ is initialized with the previous channel estimates (with each zero replaced by a small number, 10^{-5}), followed by 3 iterations of SLIM. The purpose of replacing each zero with 10^{-5} is to allow for activation of a previously assumed zero tap.

3.3 Symbol Detection

3.3.1 Detection Scheme

The general idea behind RELAX-BLAST is to detect the strongest stream first using any of the three aforementioned detection algorithms (RELAX-BLAST, V-BLAST

and LCN become identical approaches for the single transmitter case), and retrieve the source bits carried in the strongest stream through decoding; see Figure 3-2.

To make use of SIC, given the estimated source bits, we first follow the steps in the symbol generation process shown in Figure 3-1 by assuming that the structure of the encoder and interleaver is perfectly known at the receiver side: the estimated source bits are fed into the convolutional encoder, followed by a random interleaver and QPSK mapping module. This way, an error-free decoding can provide a perfect recovery of the transmitted QPSK symbols in the strongest stream. Then, the re-constructed QPSK symbols are convolved with the associated CIR estimates to yield the interference stream seen by the remaining undetected streams. The so-obtained interference stream will be subtracted out from the measurements to aid the detection of the remaining streams (this is where the interference cancellation mechanism shown in Figure 3-2 comes into play). Next, the second strongest stream (which now becomes the strongest one among the remaining $N-1$ streams since the contributions of the strongest stream have been removed) is estimated. The two detected streams are updated in an iterative manner until the estimates do not change significantly in two consecutive iterations. By iterative update, we mean that once the estimated source bits of the second strongest stream are available, their corresponding interference stream is subtracted out from the original measurements to re-determine the strongest stream. Due to the absence of the second strongest stream and hence a higher signal-to-interference-plus-noise ratio (SINR), this re-determination stage yields a better estimate of the strongest stream. This iterative update of the two strongest streams is continued until a prescribed iteration number is reached (typically 3 iterations are enough). Then, the two dominant streams are subtracted out from the measurements to estimate the third strongest stream. The first three streams are then updated iteratively by updating one stream at a time. The same procedure is repeated until all the transmitted data streams are detected and

updated. It is the iterative refinement step of RELAX-BLAST that provides performance improvements, as compared to, for instance, V-BLAST; see [48].

3.3.2 Efficient LMMSE Filtering

Let

$$\mathbf{s}_n = [\hat{\mathbf{h}}_{n,1}^T, \dots, \hat{\mathbf{h}}_{n,M}^T]^T, \quad n = 1, \dots, N, \tag{3–23}$$

denote the steering vector corresponding to $x_n(t_0)$ in (2–26). The Wiener filter corresponding to $x_n(t_0)$, denoted as $\mathbf{f}_{n,\mathcal{A}}$, is then given by [48, 91]

$$\mathbf{f}_{n,\mathcal{A}} = \mathbf{Q}_{\mathcal{A}}^{-1}\mathbf{s}_n, \quad n = 1, \dots, N, \tag{3–24}$$

where $\mathbf{s}_n$ and $\mathbf{f}_{n,\mathcal{A}} \in \mathbb{C}^{M\tilde{R}\times 1}$, and

$$\mathbf{Q}_{\mathcal{A}} = \sum_{j=1, j\notin\mathcal{A}}^{N} \hat{\mathbf{H}}_j\hat{\mathbf{H}}_j^H + \bar{\eta}\mathbf{I} \tag{3–25}$$

denotes the residue signal covariance matrix. It possibly contains some of the transmitted data streams, whose indices are collected in the set $\mathcal{A}$, subtracted out from the measurements $\tilde{\mathbf{y}}$. In our analysis, the noise power $\bar{\eta}$ in (3–25) is determined as the noise power obtained using the SLIM algorithm, see (3–13), averaged over the M receivers. The symbol estimate $\hat{x}_n(t_0)$ is then obtained by multiplying $\mathbf{f}_{n,\mathcal{A}}^H$ with the corresponding residue signal (i.e., the measurement signal with the contributions from the transmitters in $\mathcal{A}$ removed).

Due to its iterative nature, RELAX-BLAST requires computation of multiple LMMSE filters [48]. As an example, in the presence of 3 transmitters and assuming that the detection order is transmitter 1 to 3, where transmitters 1 and 3 encounter the strongest and weakest channels, respectively, RELAX-BLAST requires 6 distinct filters: $\mathbf{f}_{1,\{\emptyset\}}$, $\mathbf{f}_{2,\{1\}}$, $\mathbf{f}_{1,\{2\}}$, $\mathbf{f}_{3,\{1,2\}}$, $\mathbf{f}_{1,\{2,3\}}$ and $\mathbf{f}_{2,\{1,3\}}$. In the general case with N transmitters, $N(N+1)/2$ LMMSE filters in total are needed. The process becomes computationally expensive due to the $M\tilde{R} \times M\tilde{R}$ matrix inversion required in order to determine each individual filter. Fortunately, the conjugate gradient (CG) [62] method can be employed

in RELAX-BLAST to ease the computational burden. The pseudo-code for obtaining the LMMSE filters $\mathbf{f}_{n,\mathcal{A}}$ using the CG method is given in Table 3-1. As observed, the bottleneck in the CG implementation is the matrix-vector multiplication $\mathbf{Q}_{\mathcal{A}}\mathbf{d}^{(i)}$ required in each iteration. We use the diagonalization property of circulant matrices to replace this matrix-vector product with simple and efficient FFT operations. Our empirical experience indicates that the CG method, if not used in conjunction with the FFT implementation, does not have a noticeable advantage in terms of computational complexity over exploiting the well-known matrix inversion lemma. Specifically, the problem of interest is to find a fast way to compute $\mathbf{Q}_{\mathcal{A}}\mathbf{d}$ (the CG iteration index i is omitted for notational simplicity), or equivalently,

$$\mathbf{z}_n = \hat{\mathbf{H}}_n\hat{\mathbf{H}}_n^H\mathbf{d}, \quad n = 1, \dots, N, \quad n \notin \mathcal{A}, \tag{3–26}$$

see (3–25). Before proceeding to demonstrate how to exploit the block structure of the channel matrix $\hat{\mathbf{H}}_n$ in order to compute $\mathbf{z}_n$ efficiently, it is instructive to review the elegant diagonalization properties of a circulant matrix.

It is well-known that a $K \times K$ complex-valued circulant matrix $\mathbf{C}$ can be expressed as $\mathbf{C} = \mathbf{F}\mathbf{\Lambda}\mathbf{F}^H$. Here, $\mathbf{F}$ is the $K \times K$ FFT matrix and $\mathbf{\Lambda}$ holds the K eigenvalues of $\mathbf{C}$ along its diagonal (e.g., [71]). Furthermore, these eigenvalues can efficiently be computed by applying FFT to the first row of $\mathbf{C}$. Let $\mathbf{C}_{n,m} \in \mathbb{C}^{(2\tilde{R}-1)\times(2\tilde{R}-1)}$ denote the circulant matrix obtained by appending the appropriate $(\tilde{R}-1) \times (2\tilde{R}-1)$ matrix to the bottom of $\hat{\mathbf{H}}_{n,m}$. That is, $\mathbf{C}_{n,m}$ can be constructed by cyclically shifting the last row of $\hat{\mathbf{H}}_{n,m}$ to the right for a total of $\tilde{R}-1$ times. Then, based on the above observation, $\mathbf{C}_{n,m}$ can be diagonalized as

$$\mathbf{C}_{n,m} = \mathbf{F}\mathbf{\Lambda}_{n,m}\mathbf{F}^H, \tag{3–27}$$

for $n = 1, \dots, N$ and $m = 1, \dots, M$. Through this construction, (3–26) can be computed using $\tilde{\mathbf{z}}_n = \mathbf{C}_n\mathbf{C}_n^H\tilde{\mathbf{d}}$, where $\mathbf{C}_n \in \mathbb{C}^{M(2\tilde{R}-1)\times(2\tilde{R}-1)}$ is obtained similarly to $\hat{\mathbf{H}}_n$ in (3–26), as $\mathbf{C}_n = [\mathbf{C}_{n,1}^T \dots \mathbf{C}_{n,M}^T]^T$, and $\tilde{\mathbf{d}} \in \mathbb{C}^{M(2\tilde{R}-1)\times 1}$ is constructed by first dividing $\mathbf{d}$ into M

Table 3-1. The conjugate gradient method for RELAX-BLAST.

$\mathbf{d}^{(0)} = \mathbf{r}^{(0)} = \mathbf{s}_n$; $\tilde{\mathbf{f}}^{(0)} = \mathbf{0}$; $i = 0$	
repeat	
$\alpha^{(i)} = \frac{\|\|\mathbf{r}^{(i)}\|\|^2}{\mathbf{d}^{(i)H}\mathbf{Q}_{\mathcal{A}}\mathbf{d}^{(i)}}$	% determine the step size
$\tilde{\mathbf{f}}^{(i+1)} = \tilde{\mathbf{f}}^{(i)} + \alpha^{(i)}\mathbf{d}^{(i)}$	% update the estimate
$\mathbf{r}^{(i+1)} = \mathbf{r}^{(i)} - \alpha^{(i)}\mathbf{Q}_{\mathcal{A}}\mathbf{d}^{(i)}$	% calculate a new residual
$\mathbf{d}^{(i+1)} = \mathbf{r}^{(i+1)} + \frac{\|\|\mathbf{r}^{(i+1)}\|\|^2}{\|\|\mathbf{r}^{(i)}\|\|^2}\mathbf{d}^{(i)}$	% calculate a new search direction
$i = i + 1$	
until $\left(\frac{\|\|\mathbf{r}^{(i)}\|\|}{\|\|\mathbf{s}_n\|\|} < t_{\mathrm{CG}}\right)$	
$\mathbf{f}_{n,\mathcal{A}} = \tilde{\mathbf{f}}^{(i)}$	

groups, each of length $\tilde{R}$, and then padding $\tilde{R} - 1$ zeros between each group. Stated mathematically, if $\mathbf{d} = [\mathbf{d}_1^T, \dots, \mathbf{d}_M^T]^T$ and $\tilde{\mathbf{d}} = [\tilde{\mathbf{d}}_1^T, \dots, \tilde{\mathbf{d}}_M^T]^T$, where $\{\mathbf{d}_m\}_{m=1}^M \in \mathbb{C}^{\tilde{R}\times 1}$ and $\{\tilde{\mathbf{d}}_m\}_{m=1}^M \in \mathbb{C}^{(2\tilde{R}-1)\times 1}$, then $\tilde{\mathbf{d}}_m = [\mathbf{d}_m^T\ 0, \dots, 0]^T$ for $m = 1, \dots, M$. Furthermore, $\tilde{\mathbf{z}}_n$ is partitioned similarly as $\tilde{\mathbf{z}}_n = [\tilde{\mathbf{z}}_{n,1}^T, \dots, \tilde{\mathbf{z}}_{n,M}^T]^T$, where $\{\tilde{\mathbf{z}}_{n,m}\}_{m=1}^M \in \mathbb{C}^{(2\tilde{R}-1)\times 1}$ are computed as follows:

$$\tilde{\mathbf{z}}_{n,m} = \mathbf{C}_{n,m}\sum_{m=1}^{M}\mathbf{C}_{n,m}^H\tilde{\mathbf{d}}_m = \mathbf{F}\mathbf{\Lambda}_{n,m}\sum_{m=1}^{M}\mathbf{\Lambda}_{n,m}^H\mathbf{F}^H\tilde{\mathbf{d}}_m, \tag{3–28}$$

where $n = 1, \dots, N$ and $m = 1, \dots, M$. Finally, $\mathbf{z}_n$ in (3–26) can be obtained from the computed $\tilde{\mathbf{z}}_n$ by straightforward re-indexing: if we denote $\mathbf{z}_n = [\mathbf{z}_{n,1}^T \dots \mathbf{z}_{n,M}^T]^T$ where $\{\mathbf{z}_{n,m}\}_{m=1}^M \in \mathbb{C}^{\tilde{R}\times 1}$, then $\mathbf{z}_{n,m}$ can be obtained from the first $\tilde{R}$ elements of $\tilde{\mathbf{z}}_{n,m}$ for $m = 1, \dots, M$.

The overall complexity of this approach is reduced to $\mathcal{O}(MN\tilde{R}\log\tilde{R})$ since the $\{\mathbf{F}^H\tilde{\mathbf{d}}_m\}$ products in (3–28) are in fact nothing but efficient inverse FFT (IFFT) operations that can be computed with complexity $\mathcal{O}(\tilde{R}\log\tilde{R})$ (similarly, multiplying $\mathbf{F}$ on the left side of a vector in (3–28) is an FFT operation). Moreover, multiplying $\mathbf{\Lambda}_{n,m}$ (or $\mathbf{\Lambda}_{n,m}^H$) on the left side of a vector has a complexity of only $\mathcal{O}(2\tilde{R} - 1)$ thanks to its diagonal structure. In contrast, the computational complexity of calculating $\mathbf{Q}_{\mathcal{A}}\mathbf{d}^{(i)}$ directly is $\mathcal{O}(M^2\tilde{R}^2)$. Yet another significant advantage of the proposed fast algorithm is that it facilitates parallel processing; see (3–28). Therefore, the FFT based CG method not

only reduces the computational complexity of RELAX-BLAST considerably, but also makes RELAX-BLAST amenable to parallel implementations.

The required number of iterations needed to yield the solution in (3–24) is equal to the number of distinctive eigenvalues of $\mathbf{Q}_{\mathcal{A}}$. More often than not, however, the CG iteration can be terminated early for the sake of computational savings and possibly also for better equalization performance [25]. It was empirically observed in our experiments that it suffices to terminate the CG iterations when the ratio of the Euclidean norm of the residue vector to the Euclidean norm of the steering vector $\mathbf{s}_n$ becomes less than a predefined threshold t_{CG}. The selection of t_{CG} will be investigated in the following section.

3.4 Numerical and Experimental Results

3.4.1 Simulations

3.4.1.1 Channel estimation

We consider the problem of CIR estimation in a 4×1 multi-input single-output (MISO) system with time-invariant channels. The modulus of the simulated CIRs corresponding to the four transmitters are shown in Figure 3-3, where $\tilde{R} = 20$ delay taps are considered. The channel length is assumed to be known by the receiver and the automatic determination of channel length is thus not involved in this example. Four training sequences are synthesized by the cyclic approach (CA) [48], each of length $P = 128$. Given the CIRs and training symbols, the received measurements are constructed as in (2–5) with $\mathbf{e}_1 \sim \mathcal{CN}(\mathbf{0}, \eta\mathbf{I})$.

The mean-squared errors (MSEs) of the channel estimates obtained by IAA, SLIM and LS are shown in Figure 3-4, where each point is the average error over 100 Monte-Carlo trials. (Therefore, MSE$= \frac{1}{100}\sum_{i=1}^{100} \|\mathbf{h}_1 - \hat{\mathbf{h}}_1^{(i)}\|^2$. The combined CIR vector $\mathbf{h}_1$ of length $4\tilde{R} = 80$ has been defined in (2–5) and $\hat{\mathbf{h}}_1^{(i)}$ represents its estimate at the ith Monte-Carlo trial.) It is observed that SLIM yields slightly better MSE performance than IAA (Figure 3-4). It is also interesting to note that in this example, it is on average

3.8 times faster than IAA (note that the complexities of SLIM and IAA, as remarked in Section 3.2.2, are $\mathcal{O}(N^3\tilde{R}^3)$ and $\mathcal{O}(d_y^3)$, respectively, and in this example, $N\tilde{R} = 80$ and $d_y = P + \tilde{R} - 1 = 147$). LS gives the worst performance, mainly because it involves no mechanism to address sparsity and is a data-independent approach.

3.4.1.2 Symbol detection

Comparisons between V-BLAST and RELAX-BLAST, in terms of coded BER performance, is considered for a simulated 4×12 MIMO system. The modulus of the 4 simulated CIRs at receiver 1 are shown in Figure 3-3, and the CIRs for the other 11 receivers share similar structure. The detection order is determined as 1 (strongest), 2, 4 and 3 (weakest). We assume that channel characteristics are known to the receiver as prior knowledge and that the noise is distributed according to $\mathcal{CN}(\mathbf{0}, \eta\mathbf{I})$. In each trial, each transmitter sends one payload block (i.e., 250 coded QPSK symbols), which, per the discussion in Section 3.1, is generated by feeding 250 source bits into a 1/2-rate convolutional encoder with generator polynomials (1 0 0 1 1) and (1 1 0 1 1) followed by a random interleaver and QPSK mapping. Coded BER performance of V-BLAST and RELAX-BLAST are shown in Figure 3-5. Each point is averaged over 26000 Monte-Carlo runs. The SINR for transmitter n ($n \in \{1, 2, 3, 4\}$ in this example) is defined as

$$\mathrm{SINR} = \frac{\sum_{m=1}^{12} \|\mathbf{h}_{n,m}\|^2}{\sum_{j=1, j\neq n}^{4} \sum_{m=1}^{12} \|\mathbf{h}_{j,m}\|^2 + \eta}. \tag{3–29}$$

The binary source bits, the mapping indices of the random interleaver and the noise patterns vary independently from one trial to another. One observes from Figure 3-5 that RELAX-BLAST significantly outperforms V-BLAST. This observation is also consistent with our in-water empirical experiences. Therefore, the SPACE08 results we discuss below are all obtained using RELAX-BLAST.

3.4.2 SPACE08 In-Water Experimentation Results

3.4.2.1 Experimental specifications

The SPACE08 was conducted by WHOI at the Air-Sea Interaction Tower, 2 miles south to the coast of Martha's Vineyard, MA at a water depth of about 15 m. The MIMO system was equipped with 4 transmit transducers. The primary transmit transducer was located approximately 4 m above the bottom of the ocean using a stationary tripod. Below the primary transducer, a source array consisting of 3 transducers was deployed vertically with a spacing of 0.5 m between the elements. The top element of the source array was 3 m above the bottom of the ocean. Three separate receiver configurations were employed: 1) a 32 hydrophone cross array mounted at 60 m, 2) a 24 hydrophone vertical line array mounted at 200 m, and 3) a 12 hydrophone vertical line array mounted at 1 km. For the 60 m cross array, a horizontal leg and a vertical leg were seamed together at the center, and each leg was mounted with 16 hydrophones. The center-to-center spacing between individual elements was 3.75 cm, 5 cm and 12 cm for configurations 1-3, respectively. The top hydrophone for each configuration was approximately 3.3 m above the ocean floor. The carrier frequency and bandwidth used in the experiments were 13 KHz and 10 KHz, respectively. The symbol rate employed in SPACE08 was 7.8125 K symbols per second at each transmitter. The pulse shaping filter was a squared raised cosine filter with a rolloff factor 0.25. The baseband sampling rate was 7.8125 KHz, i.e., a symbol rate sampling scheme was adopted.

The transmitted data packets used CA training sequences of length 512 followed by QPSK payload symbols. Note that although guard intervals were not present in the experimental data (for the sake of increasing the data rate and preventing the CIR estimates from outdating), CA training sequences were still observed to provide satisfactory performance as will be shown shortly. The payload sequence of length 8 K was divided into 32 blocks, each of length 250. The generation of each 250-symbol

block follows the same procedure as elaborated in Section 3.4.1.2). The employment of 4 transducers and QPSK modulation results in a coded payload data rate of 31.25 Kbps.

The SPACE08 data encountered rich channel conditions over the course of the experiments, which is evident in Figure 3-6, where the dynamic variations of the average wave height is shown. Using the wave height as a reference, channel conditions have been roughly divided into the "ugly" category on Julian dates 300 and 301 (indicated with the "$*$" marks in Figure 3-6), "bad" category on dates 294, 295, 296, 299 and 302, and "good" category on the remaining 5 dates. The good days are chosen such that the corresponding wave height is consistently less than 1.25 m, indicated as the dash-dot line in Figure 3-6. In this way, our experimental data consists of 9 different scenarios as there are 3 receiver configurations and 3 channel conditions. Figure 3-7 shows the evolution of the normalized CIR between a given transmitter and receiver pair over time for these 9 scenarios. In these plots, a single transducer continually transmits an m-sequence, while the other transducers are inactive. One observes that the channel taps experience significant variations over time as wave height increases.

3.4.2.2 Ambient noise analysis

The sea ambient noise was modeled as a circularly symmetric complex-valued zero-mean white Gaussian random process; see (3–3). To verify the practical validity of this assumption, Figure 3-8 shows the spectral estimate of 1 km measurements acquired on Julian date 300. Figure 3-8A is obtained from 10 K samples of in-water ambient noise, while Figure 3-8B is during the data transmission where the four transmitters are transmitting signal simultaneously. Recalling that the bandwidth (or symbol rate) employed in SPACE08 in-water experiment was 7.8125 KHz, the frequency range shown in Figure 3-8 is confined to $[-3900\ 3900]$ Hz due to Nyquist sampling theory. The flat power spectrum shown in Figure 3-8A indicates that it is reasonable to approximate the ocean noise as a white Gaussian process.

3.4.2.3 Channel length selection

We provide here two experimental examples to illustrate how the channel length estimator performs. As mentioned in Section 3.2.3, a potentially large channel tap number R should initially be chosen. For SPACE08, we used $R = 100$. Consider a packet from epoch "2950157" with receiver configuration 1. The resulting ratio $\bar{\beta}_n(r)$ averaged over the 32 receivers are plotted superimposed in Figure 3-9A. Note that only the $50^{\text{th}} - 100^{\text{th}}$ taps are shown and the horizontal dash-dot line represents the threshold value 3. We observe that the tap indices at which the curves first cross the threshold are 74, 73, 71 and 70 for transmitters 1-4, respectively. Therefore, $\tilde{R}$ is determined as the minimal of the 4 candidate values, i.e., 70 in this example. Figure 3-9B shows $\bar{\beta}_n(r)$ by considering a packet from epoch "2990156" with receiver configuration 2. The same guideline determines $\tilde{R}$ as 72.

3.4.2.4 Stopping criterion for the conjugate gradient method

Recall that the CG iterations are terminated when the ratio of the Euclidean norm of the residual term to the Euclidean norm of the steering vector becomes less than t_{CG}, see Table 3-1. In this section we present guidelines on how to select t_{CG}. Consider training-directed channel estimation with channel tap number $\tilde{R} = 70$ under good channel conditions (other channel conditions lead to similar results). The average number of iterations required by the CG method for different t_{CG} settings is plotted in Figure 3-10. One observes that for $t_{\text{CG}} > 10^{-2}$, the average iteration number for 3 receiver configurations shows good agreement. It is preferable to keep the iteration number within 10 for the sake of computational efficiency. Therefore, t_{CG} is fixed at 0.05 collectively for receiver configurations 1-3. It was empirically observed in our experiments that this threshold value yields excellent BER performance. Moreover, we also noticed that the BER performance is not sensitive to the choice of t_{CG}.

3.4.2.5 Coded bit error rate performance

The channel tracking approach adopted in this paper is illustrated in Figure 3-11. In this approach, the CIR is first estimated in training-directed mode. Subsequently, based on the initial CIR estimates, the first S payload symbols (S is set to 250) are obtained using RELAX-BLAST. Next, the channels are updated in decision-directed mode using L symbols (containing the previously detected payload symbols, and possibly a portion of the training sequence as well), see Figure 3-11. Regarding the tracking length L, it is advantageous to set different L values for different receiver locations, and our empirical experience suggests to fix L at 450, 500, and 700 for receiver configurations 1-3, respectively. With the updated CIRs, the subsequent S payload symbols are detected using RELAX-BLAST. This process continues until all payload symbols are detected.

A comprehensive summary of the detection performance for receiver configuration 2 (i.e., the 4×24 MIMO system at 200 m distance) is provided in Table 3-2 based on our analysis of all 96 packets recorded over the entire course of the SPACE08 under all of the channel conditions experienced. We deem a payload data block to be successfully detected if its coded BER averaged over the 4 transmitters is less than 0.1. By adopting the proposed MIMO UAC techniques, we have succeeded in tracking the entire 32 payload blocks for 92 out of the 96 packets. A coded BER of 4.27×10^{-5} is achieved after averaging over the 7.36×10^5 source bits processed (recall that each packet carries 8000 source bits and we have 92 such packets). The resulting data rate is approximately 31.25 Kbps.

With the remaining 4 packets (2 on Julian date 300, and 2 on 301), we lost track of the channel during the payload symbol detection and channel tracking process. From Figure 3-6, it is observed that these two days (marked as "$*$") indeed experienced the most severe channel conditions. Nevertheless, it is still encouraging that we succeeded in tracking at least the first two data blocks for these 4 packets, thanks to the good auto- and cross-correlation properties of the CA training sequences used in the experiment

[48]. The coded BER of the first 2 data blocks averaged over these 4 packets is 6.25×10^{-4}. Therefore, when the channel conditions are very severe, we can consider the scheme shown in Figure 3-12, where the estimated channel is used to initialize channel equalization and decoding for 2 payload data blocks preceding and following the training sequence. In this scheme, though, the training sequences play a major role for channel estimation. Due to the training overhead, this scheme results in a data rate of 20.7 Kbps.

The receiver configurations 1 and 3 encounter more challenging channel conditions than the receiver configuration 2. To see this, Table 3-3 details the coded BER performance of receiver configuration 3. It is observed that under good channel conditions, the entire 32 data blocks for all the 48 packets can be successfully detected with an average coded BER of 3.86×10^{-4}. Under bad channel conditions, among the 30 packets processed, there are 10 for which the channel cannot be tracked. However, we still succeed tracking at least the first two data blocks for 9 out of these 10 packets. For the 18 packets transmitted under ugly channel conditions, at least the first 2 blocks of 10 of these packets were successfully detected. For the remaining 8 packets, only the first block has been successfully detected for 5 packets, and the remaining 3 packets could not be detected. Table 3-4 shows the same analysis on performance with receiver configuration 1. It can be observed that among the 3 receiver configurations considered, this particular receiver at 60 m gives the worst overall performance.

It is worth pointing out that although the surface-interactive taps at both 60 m and 200 m ranges experience significant variations (Figure 3-7), it is the difference in their power levels that account for the remarkable contrast in the resulting BER performance. The surface-interactive taps at 200 m does not possess much power, but those at 60 m are much more powerful. Under severe channel conditions, these taps are extremely difficult to track, which can severely degrade the BER performance. The more powerful these taps are, the more severe the BER degradation.

Table 3-2. 200 m performance of using sparse learning via iterative minimization (SLIM).

	Number of	All 32 blocks successful		First 2 blocks successful		First block successful	
Dates	Packets	%	average BER	%	average BER	%	average BER
good	48	100	3.26×10^{-6}	-	-	-	-
bad	30	100	7.30×10^{-6}	-	-	-	-
ugly	18	77.8	2.54×10^{-4}	22.2	6.25×10^{-4}	-	-

Table 3-3. 1 km performance of using SLIM.

	Number of	All 32 blocks successful		First 2 blocks successful		First block successful	
Dates	Packets	%	average BER	%	average BER	%	average BER
good	48	100	3.86×10^{-4}	-	-	-	-
bad	30	66.7	1.04×10^{-3}	30	1.93×10^{-2}	-	-
ugly	18	0	-	55.6	1.48×10^{-2}	27.8	6.42×10^{-2}

Table 3-4. 60 m performance of using SLIM.

	Number of	All 32 blocks successful		First 2 blocks successful		First block successful	
Dates	Packets	%	average BER	%	average BER	%	average BER
good	48	54.2	7.02×10^{-4}	31.3	6.20×10^{-3}	8.3	5.20×10^{-2}
bad	30	60	1.61×10^{-3}	23.3	4.93×10^{-3}	6.7	3.00×10^{-2}
ugly	18	5.6	3.13×10^{-4}	27.8	4.80×10^{-3}	11.1	4.90×10^{-2}

Table 3-5. 200 m performance of using lease squares (LS).

	Number of	All 32 blocks successful		First 2 blocks successful		First block successful	
Dates	Packets	%	average BER	%	average BER	%	average BER
good	48	97.9	2.53×10^{-5}	2.1	2.00×10^{-3}	-	-
bad	30	100	7.81×10^{-5}	-	-	-	-
ugly	18	22.2	0	77.8	2.57×10^{-3}	-	-

Table 3-6. 1 km performance of using LS.

	Number of	All 32 blocks successful		First 2 blocks successful		First block successful	
Dates	Packets	%	average BER	%	average BER	%	average BER
good	48	97.9	5.57×10^{-4}	2.1	5.00×10^{-4}	-	-
bad	30	56.7	2.52×10^{-4}	43.3	1.58×10^{-2}	-	-
ugly	18	0	-	55.6	1.87×10^{-2}	22.2	3.38×10^{-2}

Table 3-7. 60 m performance of using LS.

	Number of	All 32 blocks successful		First 2 blocks successful		First block successful	
Dates	Packets	%	average BER	%	average BER	%	average BER
good	48	6.3	1.26×10^{-3}	75.0	4.86×10^{-3}	14.6	5.36×10^{-2}
bad	30	23.3	9.15×10^{-4}	53.3	7.88×10^{-3}	13.3	3.30×10^{-2}
ugly	18	5.6	1.16×10^{-3}	11.1	1.50×10^{-3}	16.7	3.10×10^{-2}

Finally, we analyze the SPACE08 in-water experimental data using LS as channel estimation algorithms instead of SLIM, and the empirical BER results are summarized in Tables 3-5∼3-7. Unlike the SLIM algorithm, LS does not possess the capability of automatically determining the channel tap number, and therefore, the tap number of each packet determined by SLIM is used here for fair comparison. Comparing the LS results with the SLIM results summarized in Tables 3-2∼3-4, one observes that SLIM outperforms LS significantly, especially under challenging channel conditions. More specifically, at 200 m range under ugly channel conditions, among the 18 packages processed, SLIM successfully detects the entire payload sequence for 14 packages, compared to 4 packages by LS. Moreover, at 60 m range under good and bad channel conditions, SLIM detects the entire payload sequence for significantly more packages than LS.

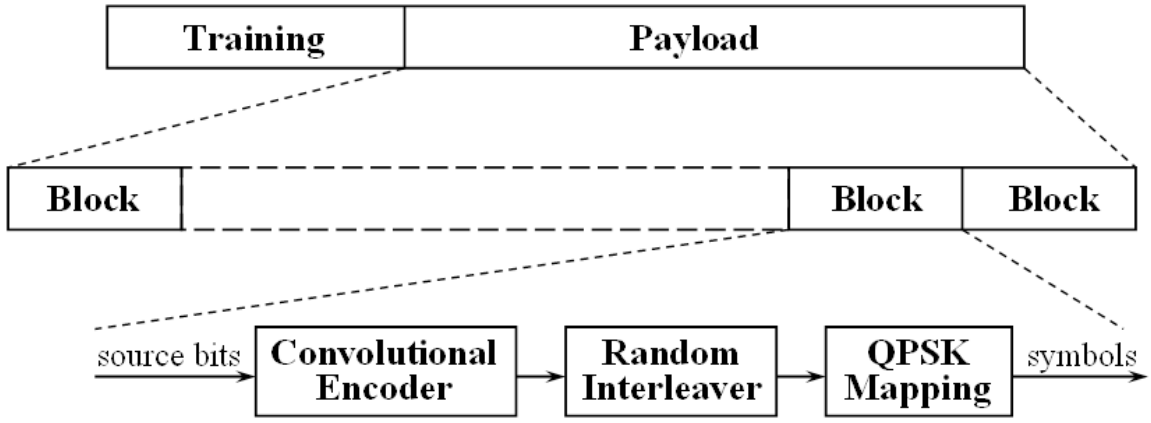


Figure 3-1. Each packet consists of a training sequence followed by a payload sequence. Each payload sequence is divided into blocks of equal length and each block is encoded separately.

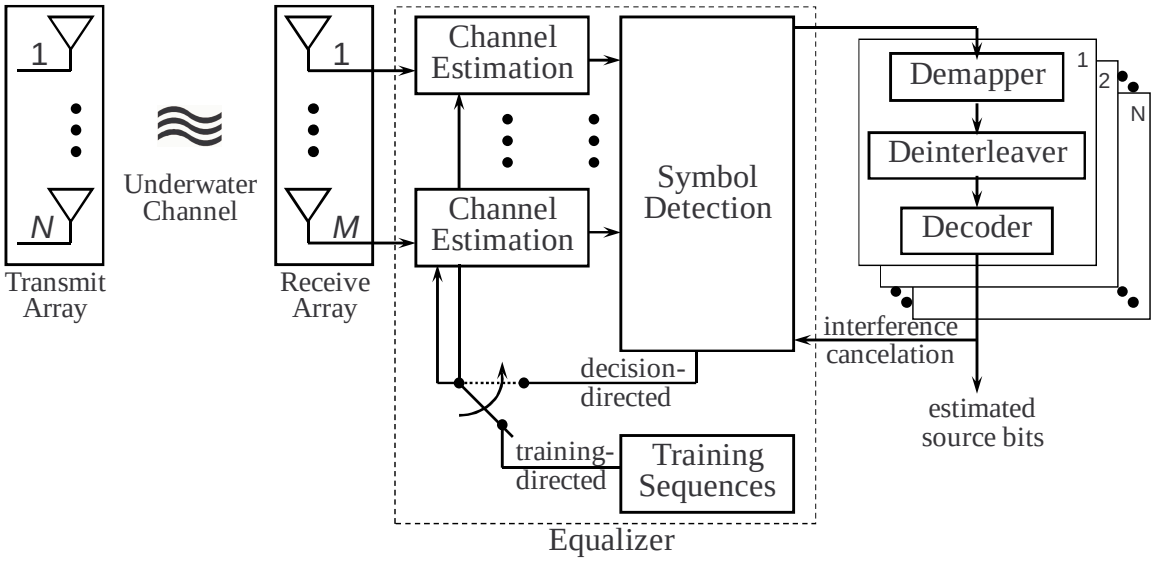


Figure 3-2. An $N \times M$ MIMO UAC system with N transmit transducers and M receive transducers.

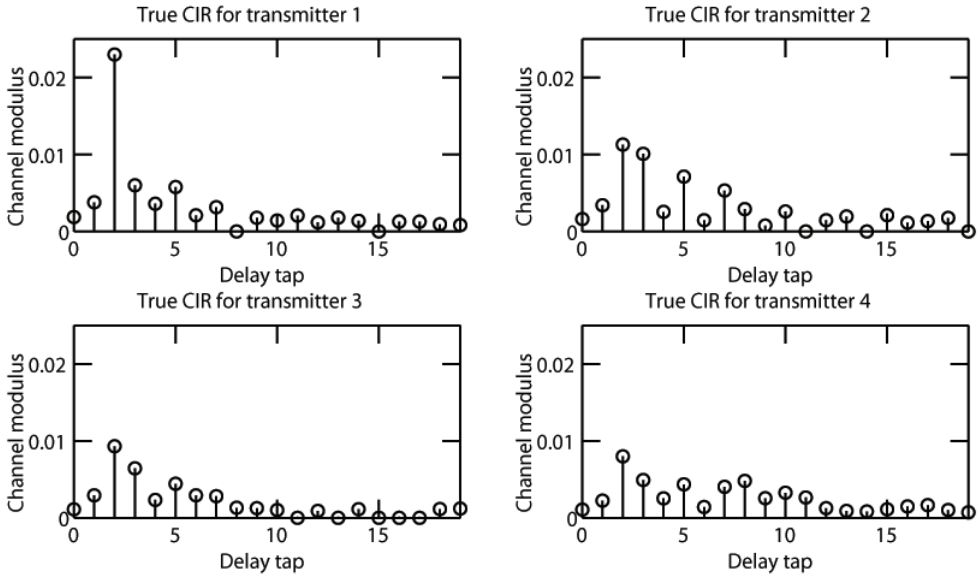


Figure 3-3. The modulus of the simulated CIRs between the four transmitters and the receiver.

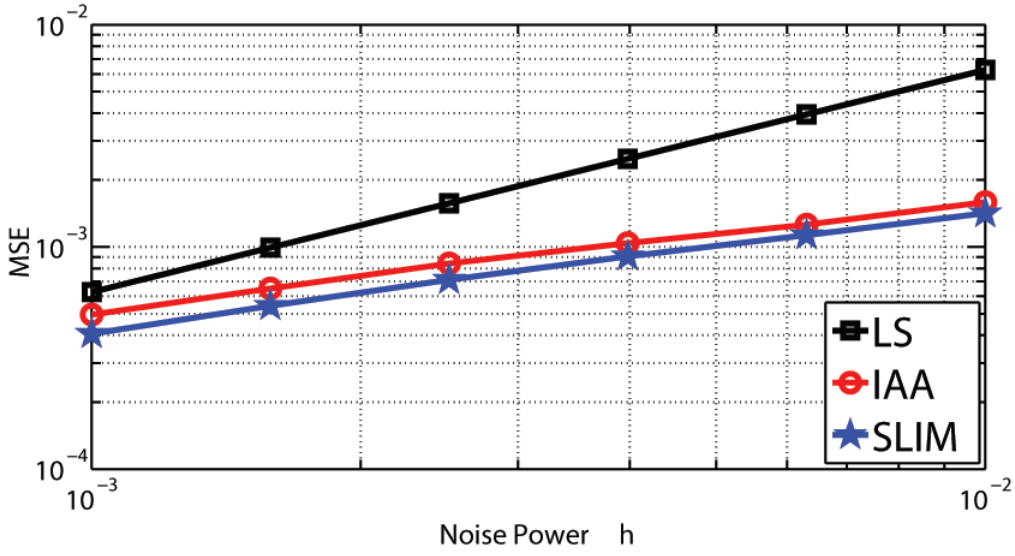


Figure 3-4. MSEs of the CIR estimates using CA training sequences of length $P = 128$ symbols. Each point is averaged over 100 Monte-Carlo trials.

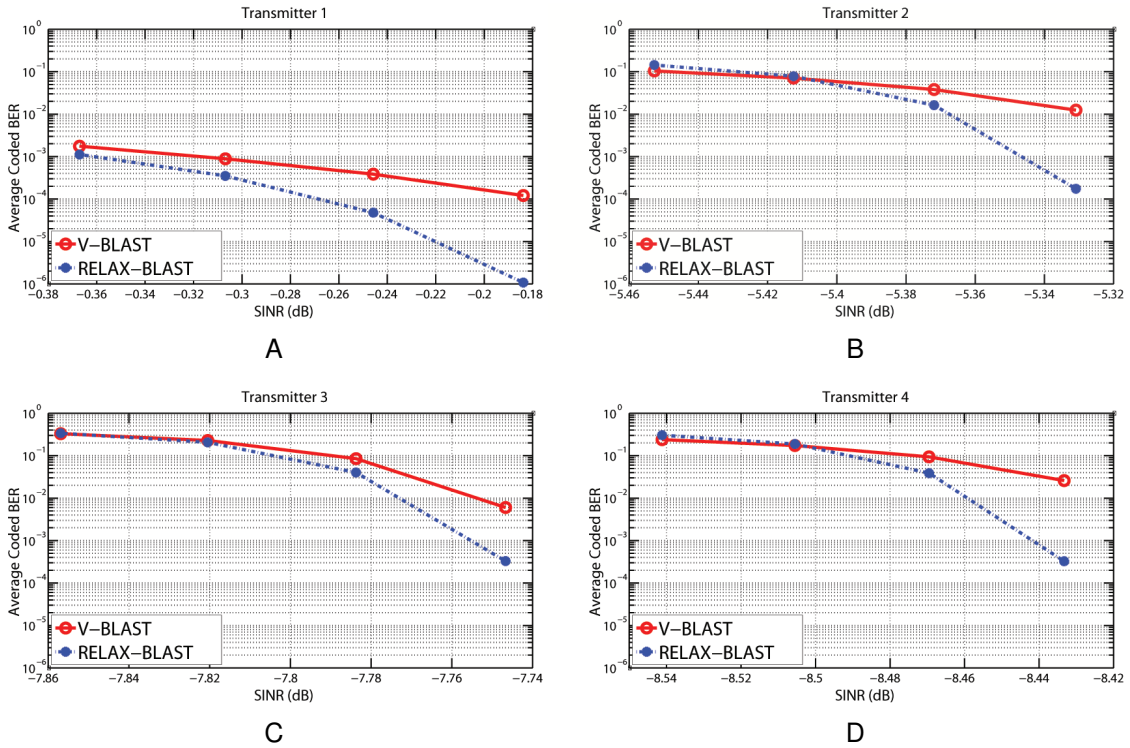


Figure 3-5. The coded BERs of each of the four transmitted payload sequences for a 4×12 MIMO system assuming the receiver has perfect knowledge on the CIR characteristics. The detection performances of V-BLAST and RELAX-BLAST are compared in terms of coded BER averaged over 26000 Monte-Carlo trials for varying levels of the noise power η.

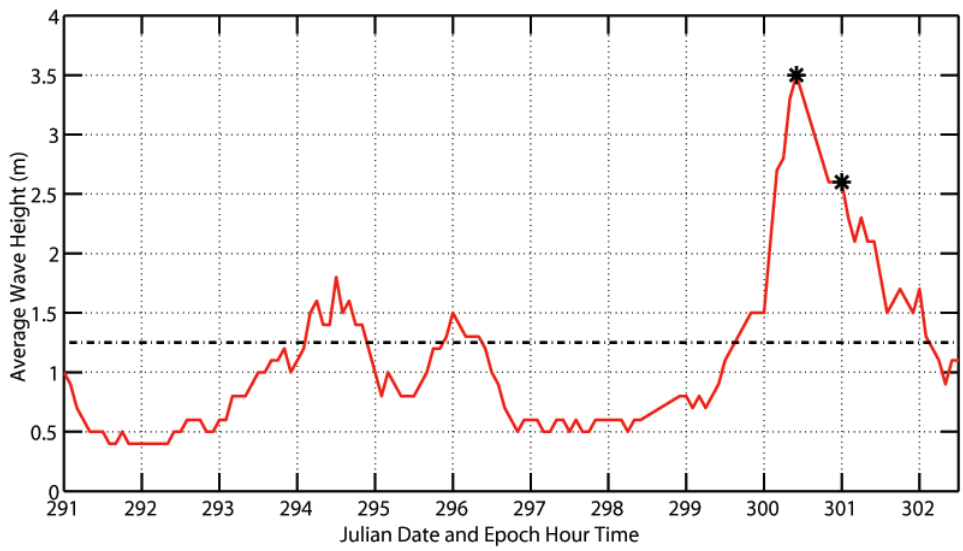


Figure 3-6. SPACE08 meteorological data. The average wave height (m) is shown over the course of the experiments. Two "$*$" marks denote Julian dates 300 and 301, which experienced the ugly channel conditions. A wave height of 1.25 m, represented as the dash-dot line, divides the remaining 10 Julian dates into two categories, namely bad channel conditions on Julian dates 294, 295, 296, 299 and 302, and good channel conditions on Julian dates 291, 292, 293, 297 and 298.

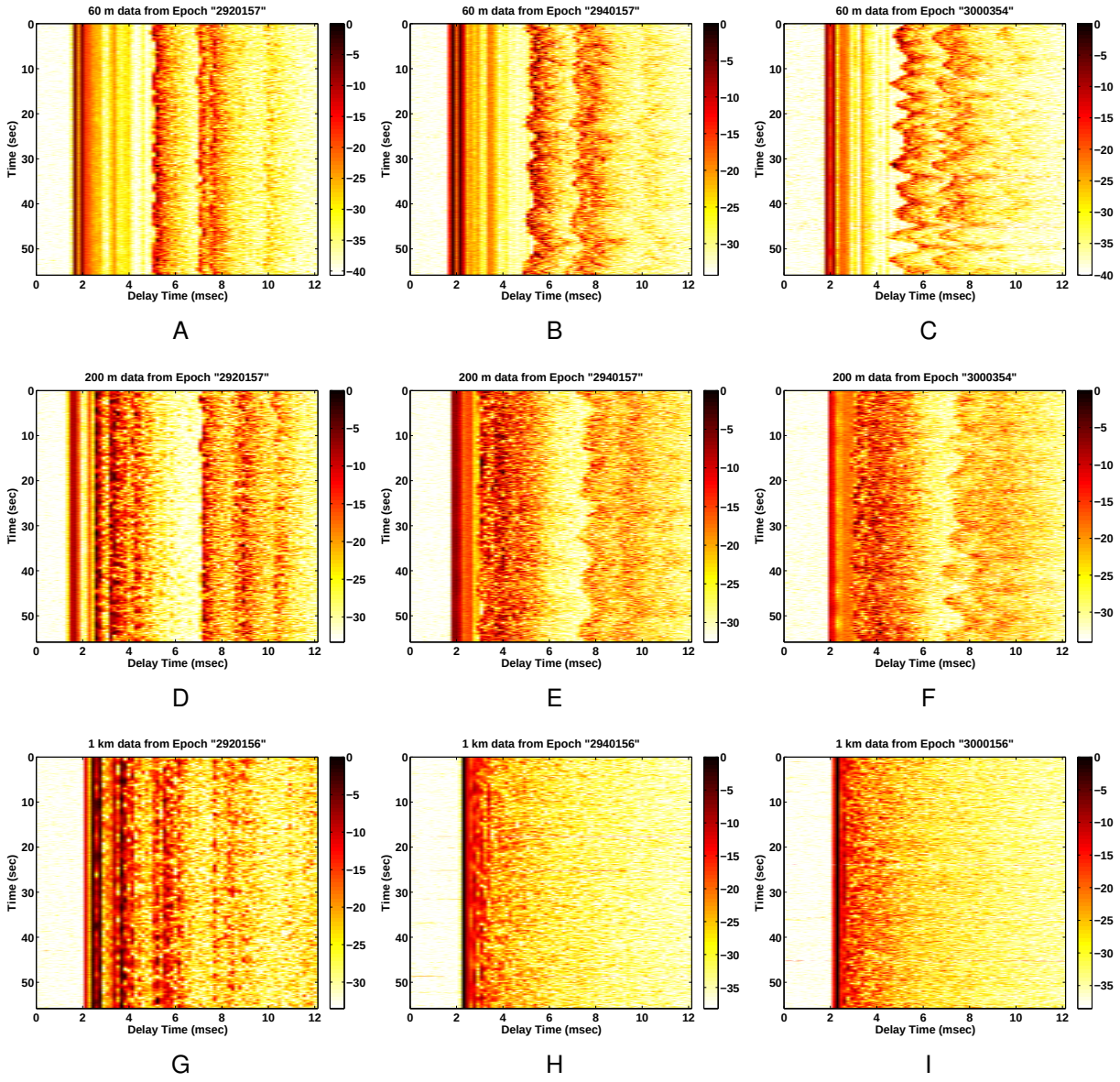
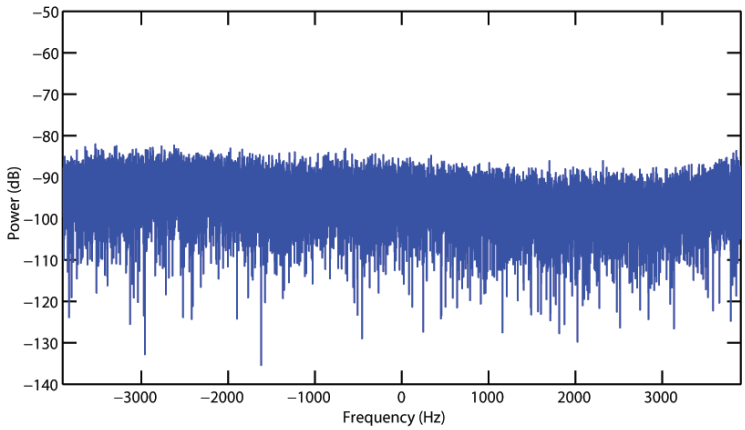
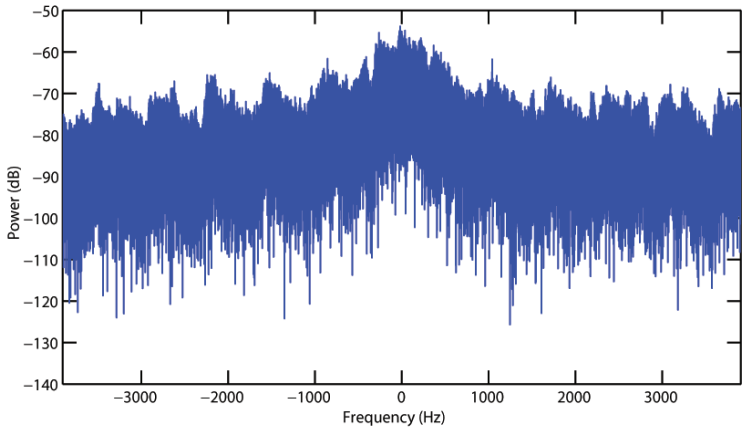


Figure 3-7. Normalized CIR evolution over approximately a 1 min period. The modulus of the channel tap is shown in dB. CIR is estimated using m-sequences. A)-C) good, bad and ugly channel conditions at 60 m. B)-F) good, bad and ugly channel conditions at 200 m. G)-I) good, bad and ugly channel conditions at 1 km. A)D)G) measurements recoded on Julian date 292. B)E)H) measurements recorded on Julian date 294. C)F)I) measurements recorded on Julian date 300.

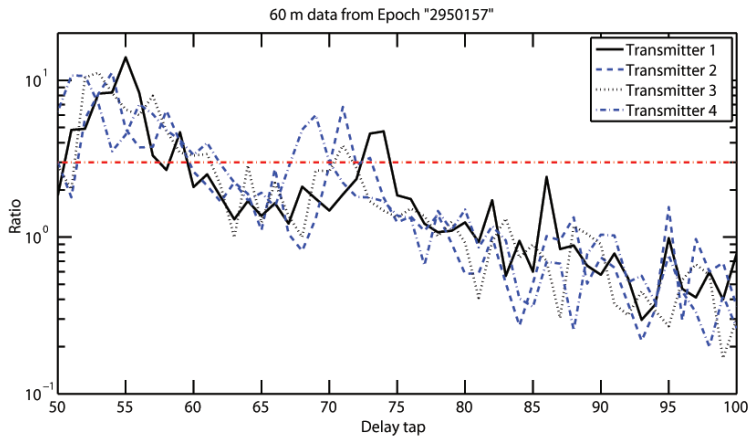


A

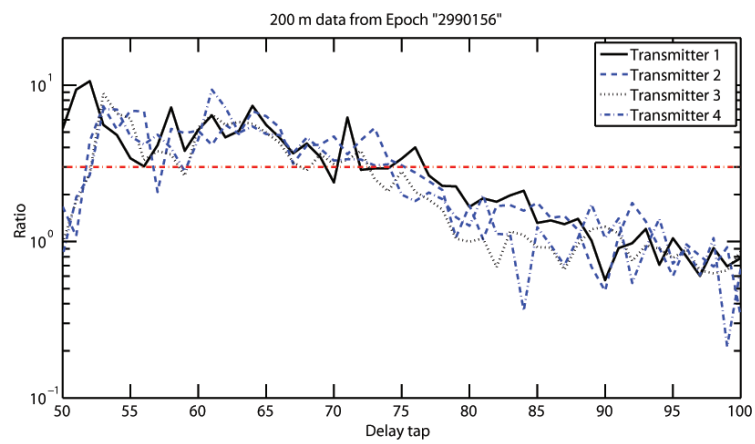


B

Figure 3-8. Spectral estimation of the received measurements at 1 km distance on Julian date 300. (A) The ambient sea noise. (B) The simultaneously transmitted signal on top of the sea noise.



A



B

Figure 3-9. The plot of $\bar{\beta}_n(r)$ versus the channel taps. Only the last 51 taps are shown and the horizontal dash-dot line represents the threshold value 3. A) considering a packet from epoch “2950157” with receiver configuration 1, and B) considering a packet from epoch “2990156” with receiver configuration 2.

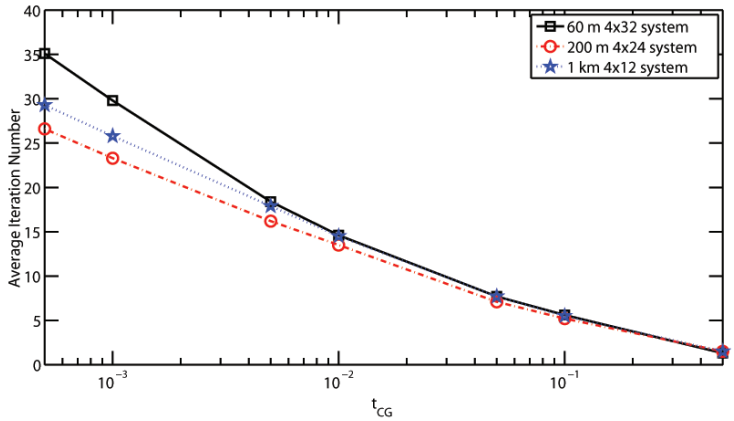


Figure 3-10. The impact of t_{CG} on the average number of iterations required by the conjugate gradient method.

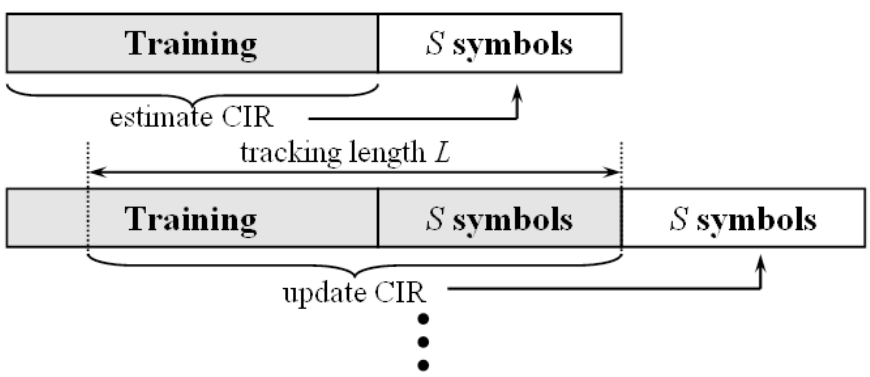


Figure 3-11. The channel tracking procedure employed in our analysis.

Figure 3-12. Data packet structure.

CHAPTER 4
MIMO UAC OVER SPARSE AND FREQUENCY MODULATED ACOUSTIC CHANNELS

So far, we have studied UAC over ISI acoustic channels by ignoring the Doppler effects. In this chapter, we incorporate the Doppler effects by discussing the coherent MIMO UAC over double spreading acoustic channels (i.e., the channels subject to both ISI and Doppler effects) coupled with time-varying characteristics. Both channel estimation and symbol detection will be discussed, but with more emphasis on the former. As mentioned in Section 1.1, a preferable tool to characterize a double spreading channel is the scattering function (SF), which essentially decouples the acoustic channel into a bank of paths that experience different delays and Doppler frequencies [43]. Due to large degrees of freedom, obtaining SF, more often than not, is equivalent to solving an underdetermined problem, where the number of unknowns is much larger than that of equations [100] (this is particularly true in the MIMO context since the SFs for all the transmit transducers should be estimated simultaneously). Since an unconstrained underdetermined problem in principle admits an infinite number of solutions, sparsity requirements should be imposed on the solution [82]. By sparsity, we mean that the underwater channel consists of only a few dominant delay and Doppler taps, while all the remaining taps can be approximated as zero [47, 100]. A detailed treatment of popular algorithms to solve such sparsity-based channel estimation problem is presented in [100], along with a thorough comparison of their performances.

The major concern in SF-based channel estimation, as previously remarked, is that the problem becomes over parameterized (i.e., too many degrees of freedom). It is practically more beneficial to look for a channel model with the smallest number of parameters, but one that still sufficiently reflects the defining characteristics of the acoustic channel of interest. To this end, an alternative method to SF (but still related to SF) is proposed in [65], where the contribution of Doppler effects comes through one common frequency value for all the transmitters. The number of unknowns, as a

consequence, is significantly reduced. While no further accounts are provided in [65] as to at what channel conditions such simplification becomes valid, we will show via experimental results presented later on that the assumption of a common Doppler frequency is generally valid when the Doppler effects are induced by the relative motions between the transmitter and receiver arrays. In [65], the channel estimation is performed in two separate steps. That is, the CIRs and the underlying Doppler frequency are estimated in a separate manner. It is advantageous to estimate both the CIRs and Doppler frequency jointly. This motivates the introduction in this chapter of an extension of the SLIM algorithm (SLIM, as proposed in Section 3.2.2, was developed for ISI channels), referred to as generalization of SLIM (GoSLIM). The fundamentals of the sparse approach is established based on a hierarchical Bayesian model, solved through maximizing the a posteriori probability (MAP). Like SLIM, GoSLIM is also user parameter free, making it easy to use in practice.

Another important aspect of UAC is symbol detection given the estimates of CIRs and Doppler frequencies. The adverse phase shift induced by the channel should first be compensated for using the so-obtained Doppler frequency estimate [65]. Such phase compensation task effectively converts the original double spreading channel to an ISI channel, which allows for the employment of various techniques that effectively combat ISI, including the Alamouti diversity [44] and Bell Labs Layered Space-Time (BLAST) type of schemes [47]. The Alamouti diversity scheme is practically attractive since it fully extracts the transmit diversity [84] and allows for efficient symbol detection [44]. Early attempts of employing the Alamouti diversity scheme in the UAC regime were reported in [53, 86]. The scheme adopted in [53] is a direct implementation of the original Alamouti idea that works well for flat-fading channels only [2]. In [86], the extended Alamouti scheme to ISI channels [44] is used to implement intermediate nodes to relay the signal between source and destination, and its effectiveness is verified using simulated data. In the present paper, we will show via in-water experimentation

results that, as long as the Doppler effects are sufficiently compensated for, the Alamouti diversity scheme is a promising candidate in UAC systems operating at a moderate data rate. To pursue a high data rate, RELAX-BLAST is also considered in our MIMO experiment.

The rest of this chapter is organized as follows. Section 4.1 formulates the channel estimation problem in both training-directed and decision-directed modes. Then we propose the GoSLIM algorithm for jointly estimating CIRs and Doppler frequencies. Section 4.2 briefly overviews the symbol detection process. Section 4.3 presents the simulation results of GoSLIM, followed by the experimental results obtained from analyzing the WHOI09 and ACOMM10 experiment data.

4.1 Channel Estimation

In this section, we start with the problem formulation of double spreading channel estimation in both training-directed and decision-directed modes. Then, we present the GoSLIM algorithm for jointly estimating the underlying CIRs and Doppler frequencies. Consider a MIMO UAC system equipped with N transmit transducers and M receive hydrophones. In what follows, unless otherwise stated, it is assumed that at each receiver the channel taps for all the N transducers experience the same Doppler frequency, but different receive hydrophones experience different Doppler shifts.

4.1.1 Training-Directed Mode

As usual, the initial task of the receiver is to acquire knowledge of the underlying channel between all transmitter and receiver pairs using the training sequences. By adopting the cyclic prefix scheme in [84], the training sequence at the nth transmitter ($n = 1, 2, \ldots, N$) is given by

$$\mathbf{x}_n = [\underbrace{x_n(P - L_{\text{CP}} + 1), \ldots, x_n(P)}_{L_{\text{CP}} \text{ prefix symbols}}, \underbrace{x_n(1), x_n(2), \ldots, x_n(P)}_{P \text{ core training symbols}}], \tag{4–1}$$

where $[x_n(1), \ldots, x_n(P)]$ is the core training sequence and the leading L_{CP} symbols form the cyclic prefix. In general, we have $P > L_{\text{CP}} \geq R - 1$, with R being the channel

tap number. From an amplifier efficiency point of view, it is practically desirable to use unimodular training sequences, i.e., $|x_n(p)| = 1$ for $n = 1, \dots, N$ and $p = 1, \dots, P$.

For MIMO UAC over double spreading channels, the measurement vectors can be written as [43, 56]

$$\mathbf{y}_m = \mathbf{\Lambda}_m \sum_{n=1}^{N} \tilde{\mathbf{X}}_n \mathbf{h}_{n,m} + \mathbf{e}_m, \quad m = 1, \dots, M, \tag{4–2}$$

where $\mathbf{y}_m$ is defined in (3–2) and $\mathbf{h}_{n,m}$ is given in (2–3). In (4–2), $\mathbf{e}_m$ represents additive noise (thermal or hardware related noise, as well as the ambient sea noise) at the mth receiver. By employing the cyclic prefix scheme, $\tilde{\mathbf{X}}_n \in \mathbb{C}^{P \times R}$ in (4–2) is given by

$$\tilde{\mathbf{X}}_n = \begin{bmatrix} x_n(1) & x_n(P) & \dots & x_n(P-R+2) \\ x_n(2) & x_n(1) & \dots & x_n(P-R+3) \\ \vdots & \vdots & \ddots & \vdots \\ x_n(P) & x_n(P-1) & \dots & x_n(P-R+1) \end{bmatrix}, \quad n = 1, \dots, N. \tag{4–3}$$

The cyclic prefix scheme entrusts $\tilde{\mathbf{X}}_n$ with a cyclic shift property, i.e., the rth column of $\tilde{\mathbf{X}}_n$ can be derived by cyclically rotating the first column by $r-1$ symbols for $r = 2, \dots, R$. It is worth pointing out that the structure of $\tilde{\mathbf{X}}_n$ should reflect the characteristics of the training sequences, as well as how we design the transmitted signal. Specifically, for CA sequences with good aperiodic correlation properties coupled with the guard interval between the training and payload sequence, $\tilde{\mathbf{X}}_n$ is given in (2–4). For the same sequences without the guard interval, $\tilde{\mathbf{X}}_n$ becomes (3–1). When employing training sequences with good periodic correlation properties coupled with the cyclic prefix scheme, as considered herein, $\tilde{\mathbf{X}}_n$ is constructed according to (4–3). Although the same matrix $\tilde{\mathbf{X}}_n$ is used in three different scenarios, the structure of $\tilde{\mathbf{X}}_n$ should be clear from the context. The so-called Doppler shift matrix $\mathbf{\Lambda}_m \in \mathbb{C}^{P \times P}$ in (4–2) is constructed as:

$$\mathbf{\Lambda}_m = \text{diag}\left(\left[1, e^{-2j\pi f_m T_s}, \ \dots \ , e^{-2j\pi f_m T_s (P-1)}\right]\right), \quad m = 1, \dots, M, \tag{4–4}$$

where T_s represents the symbol period.

The ISI and Doppler shift effects can be viewed separately in (4–2). More specifically, the term $\sum_{n=1}^{N} \tilde{\mathbf{X}}_n \mathbf{h}_{n,m}$ indicates the net contribution of N ISI channels, while the impact of the Doppler effects on the measurements comes through $\mathbf{\Lambda}_m$ only, which corresponds to the assumption that at the mth receiver, all the NR CIR taps involved (recall that we have N transmit transducers and each transducer corresponds to an R-tap channel) experience the same Doppler frequency f_m. The purpose of setting the first diagonal element of $\mathbf{\Lambda}_m$ to 1 (Equation (4–4)) is to eliminate ambiguities. In our example, relative to $\{y_m(1)\}$, a generic measurement, say $y_m(p)$, experiences a phase shift of $-f_m T_s(p-1)$.

We express (4–2) in a more compact form:

$$\mathbf{y}_m = \mathbf{\Lambda}_m \mathbf{X} \mathbf{h}_m + \mathbf{e}_m, \tag{4–5}$$

where $\mathbf{X} = [\tilde{\mathbf{X}}_1, \ldots, \tilde{\mathbf{X}}_N]$ and $\mathbf{h}_m = [\mathbf{h}_{1,m}^T, \ldots, \mathbf{h}_{N,m}^T]^T$. Then the training-directed channel estimation reduces to estimating $\mathbf{h}_m$ and f_m from the measurement vector $\mathbf{y}_m$ and known $\mathbf{X}$ for $m = 1, \ldots, M$. Equation (4–4) implicitly assumes that the channels $\{\mathbf{h}_{n,m}\}$ and the frequencies $\{f_m\}$ remain constant over the length of $\{\mathbf{y}_m\}$ (i.e., during the duration of the training sequences) and that the transmitted signals are not scaled (stretched or compressed) over the length of $\{\mathbf{y}_m\}$. Note that (4–2) includes an ISI channel estimation problem as a special case by setting $f_m = 0$. The subject of synthesizing unimodular training sequences, coupled with the employment of the cyclic prefix scheme, to facilitate ISI channel estimation is thoroughly treated in [46]. The shifted PeCAN waveforms [26] are used as the training sequences in the WHOI09 and ACOMM10 in-water experimentations.

4.1.2 Decision-Directed Mode

The decision-directed channel estimation problem is only a slight twist of its training-directed counterpart. For the former, we use the previously estimated payload symbols, instead of the training symbols, to estimate the channels. Accordingly, (4–2)

can still be used, where $\mathbf{y}_m$ and $\tilde{\mathbf{X}}_n$ are defined in (2–6) and (2–7), respectively. To conform with the matrix dimensions, the Doppler shift matrix $\mathbf{\Lambda}_m$ now is L by L (the tracking length L is defined after (2–7)), constructed as $\mathbf{\Lambda}_m = \mathrm{diag}\left(\left[1, e^{-2j\pi f_m T_s}, \ \dots \ , e^{-2j\pi f_m T_s(L-1)}\right]\right)$ for $m = 1, \dots, M$. Similarly to the training-directed mode, the channel estimation problem in the decision-directed mode aims to estimate $\mathbf{h}_m$ and f_m from the measurement vector $\mathbf{y}_m$ and known $\mathbf{X}$ formed from the decision-directed $\{\tilde{\mathbf{X}}_n\}_{n=1}^{N}$ in (2–7), for $m = 1, \dots, M$, see (4–5).

4.1.3 Channel Estimation Algorithm

Similarly to (3–3), the channel estimation algorithm at each receiver, in either training- or decision-directed mode, has the generic form given by (Equation (4–5))

$$\mathbf{y} = \mathbf{\Lambda X h} + \mathbf{e}. \tag{4–6}$$

We can see that on top of (3–3), (4–6) incorporates the Doppler shift matrix $\mathbf{\Lambda}$, which brings one more estimate target, namely the Doppler frequency f. Furthermore, by considering a flat prior on f, η and $\{p_n\}_{n=1}^{NR}$, the channel vector $\mathbf{h}$, Doppler frequency f, the covariance matrix $\mathbf{P}$ (or more precisely, its diagonal elements $\mathbf{p}$) and the noise power η can be estimated based on the MAP criterion:

$$\max_{\mathbf{h},\mathbf{p},\eta,f} p(\mathbf{h}, \mathbf{p}, \eta, f|\mathbf{y}) = \max_{\mathbf{h},\mathbf{p},\eta,f} p(\mathbf{y}|\mathbf{h}, \eta, f)p(\mathbf{h}|\mathbf{p}). \tag{4–7}$$

By combining (3–4), (3–5) and (4–7), and by taking the negative logarithm of the cost function, the optimization problem formulated in (4–7) becomes

$$\min_{\mathbf{h},\mathbf{p},\eta,f} \left(d_y \log \eta + \frac{\|\mathbf{y} - \mathbf{\Lambda X h}\|^2}{\eta} + \sum_{n=1}^{NR} \log p_n + \sum_{n=1}^{NR} \frac{|h_n|^2}{p_n} \right), \tag{4–8}$$

which, in the same spirit of SLIM, can be solved using an alternating approach.

The 5 steps of the GoSLIM algorithm at the tth iteration are outlined below:

1. Given $\mathbf{h}^{(t-1)}$, the optimal $\mathbf{P}^{(t)}$ that minimizes the cost function in (4–8) is given by:

$$p_n^{(t)} = \left|h_n^{(t-1)}\right|^2, \qquad n = 1, \dots, NR. \tag{4–9}$$

For better numerical stability, we set $p_n^{(t)}$ (or equivalently $h_n^{(t)}$) to zero if $p_n^{(t)} < 10^{-15}$. (Note that the GoSLIM algorithm achieves sparsity due to the hierarchical Bayesian model, not through this "compare-and-null" step.)

2. Once $\mathbf{P}^{(t)}$ is available, we proceed to update the CIR vector as:

$$\begin{aligned}\mathbf{h}^{(t)} &= \left[\left(\mathbf{\Lambda}^{(t-1)}\mathbf{X}\right)^H\left(\mathbf{\Lambda}^{(t-1)}\mathbf{X}\right)+\eta^{(t-1)}\left(\mathbf{P}^{(t)}\right)^{-1}\right]^{-1}\left(\mathbf{\Lambda}^{(t-1)}\mathbf{X}\right)^H\mathbf{y}\\ &= \left[\mathbf{X}^H\mathbf{X}+\eta^{(t-1)}\left(\mathbf{P}^{(t)}\right)^{-1}\right]^{-1}\left(\mathbf{\Lambda}^{(t-1)}\mathbf{X}\right)^H\mathbf{y}. \qquad (4\text{–}10)\end{aligned}$$

The second equality follows from the fact that $\mathbf{\Lambda}^{(t-1)}$ is unitary. While inverting $\mathbf{P}^{(t)}$, its zero diagonal entries are removed, and the associated columns in $\mathbf{X}$ are discarded.

3. Next, using the most recently obtained $\mathbf{h}^{(t)}$ in (4–10), we estimate the Doppler frequency f. For ease of exposition, we denote $z^{(t)}(i) = y^*(i)\tilde{x}^{(t)}(i)$, where $y(i)$ and $\tilde{x}^{(t)}(i)$ represent, respectively, the ith element of the measurement vector $\mathbf{y}$ and $\tilde{\mathbf{x}}^{(t)}$ with $\tilde{\mathbf{x}}^{(t)} = \mathbf{X}\mathbf{h}^{(t)}$, $i = 1, \ldots, d_y$. It is easy to verify that

$$\begin{aligned}\left\|\mathbf{y}-\mathbf{\Lambda}\mathbf{X}\mathbf{h}^{(t)}\right\|^2 &= \left\|\mathbf{y}-\mathbf{\Lambda}\tilde{\mathbf{x}}^{(t)}\right\|^2\\ &= \text{const}-2\text{Re}\left(\sum_{i=1}^{d_y} z^{(t)}(i)e^{-2j\pi fT_s(i-1)}\right). \qquad (4\text{–}11)\end{aligned}$$

Since the constant term in (4–11) is not a function of f, minimizing the cost function in (4–8) is equivalent to solving

$$f^{(t)} = \arg\max_f \text{Re}\left(\sum_{i=1}^{d_y} z_i^{(t)} e^{-2j\pi fT_s(i-1)}\right). \qquad (4\text{–}12)$$

Note that the summation term within the parenthesis above is nothing but the discrete-time Fourier transform (DTFT) of the sequence $\{z_i\}_{i=1}^{d_y}$ evaluated at frequency f. Therefore, $f^{(t)}$ is obtained as the location of the dominant peak of the real part of the DTFT. In practice, to ensure the accuracy of the estimate, $\{z_i\}_{i=1}^{d_y}$ should be zero-padded and then transformed by using the fast Fourier transform (FFT). We zero-pad the sequence to a length of 2^{20} in our examples.

4. Using the $\mathbf{h}^{(t)}$ and $\mathbf{\Lambda}^{(t)}$ most recently obtained via (4–10) and (4–12), respectively, we finally estimate the noise power as:

$$\eta^{(t)} = \frac{1}{d_y}\left\|\mathbf{y}-\mathbf{\Lambda}^{(t)}\mathbf{X}\mathbf{h}^{(t)}\right\|^2. \qquad (4\text{–}13)$$

5. Set $t = t+1$. Go back to Step 1 if t is less than a predefined iteration number, or terminate otherwise.

In the training-directed mode, the channel characteristics are in general not available a priori. In our examples, $\mathbf{h}^{(0)}$ is initialized using the standard matched filter, $f^{(0)}$ is initialized as 0 and the noise power $\eta^{(0)}$ is initialized with a small positive number, for instance, 10^{-10}. Our empirical experience suggests that the GoSLIM algorithm does not provide significant performance improvements after 20 iterations or less.

4.2 Symbol Detection

In this section, we focus on determining the payload symbols given the GoSLIM estimates of CIRs and Doppler frequencies. The symbol detection task is achieved via two steps: phase compensation followed by ISI equalization. We first present the problem formulation, and then, describe the phase compensation scheme. Finally, the Alamouti diversity scheme is briefly reviewed.

4.2.1 Problem Formulation

Treating the transmitted symbols as the unknowns and the CIRs and Doppler frequencies as known in (4–2), the measurement vector can be expressed as [43, 56]:

$$\mathbf{y}_m = \hat{\mathbf{\Lambda}}_m \sum_{n=1}^{N} \hat{\mathbf{H}}_{n,m} \dot{\mathbf{x}}_n + \mathbf{e}_m, \quad m = 1, \dots, M, \tag{4–14}$$

where the estimated CIR matrix $\hat{\mathbf{H}}_{n,m} \in \mathbb{C}^{R\times(2R-1)}$ is given in (2–23) while $\dot{\mathbf{x}}_n$ and $\mathbf{y}_m$ are defined in (2–24). Per the discussions following (4–4), once the estimate of Doppler frequency $\hat{f}_m$ is available, the estimated Doppler shift matrix $\hat{\mathbf{\Lambda}}_m$ in (4–14) is constructed as:

$$\hat{\mathbf{\Lambda}}_m = \mathrm{diag}\left(\left[e^{-2j\pi\hat{f}_m T_s(t_0-1)}, \ \dots \ , e^{-2j\pi\hat{f}_m T_s(t_0+R-2)}\right]\right). \tag{4–15}$$

When detecting symbols, the estimates $\{\hat{\mathbf{h}}_{n,m}\}$ and $\{\hat{f}_m\}$ are assumed fixed since the previous channel update and we treat $\{\hat{\mathbf{H}}_{n,m}\}$ and $\{\hat{\mathbf{\Lambda}}_m\}$ in (4–14) as known.

4.2.2 Phase Compensation

This task is simply achieved by multiplying $\hat{\mathbf{\Lambda}}_m^H$ to both sides of (4–14), yielding

$$\tilde{\mathbf{y}}_m = \sum_{n=1}^{N} \hat{\mathbf{H}}_{n,m} \dot{\mathbf{x}}_n + \tilde{\mathbf{e}}_m, \quad m = 1, \dots, M, \tag{4–16}$$

where $\tilde{\mathbf{y}}_m = \hat{\mathbf{\Lambda}}_m^H \mathbf{y}_m$ and $\tilde{\mathbf{e}}_m = \hat{\mathbf{\Lambda}}_m^H \mathbf{e}_m$. Given $\mathbf{e}_m \sim \mathcal{CN}(\mathbf{0}, \eta\mathbf{I})$, $\tilde{\mathbf{e}}_m$ still has the distribution of $\mathcal{CN}(\mathbf{0}, \eta\mathbf{I})$ since $\hat{\mathbf{\Lambda}}_m^H$ is unitary. Phase compensation effectively converts the original double spreading channel to an ISI channel. In (4–16), given the phase-compensated measurement vectors $\{\tilde{\mathbf{y}}_m\}$ and the estimated CIR matrices $\{\hat{\mathbf{H}}_{n,m}\}$, the task of detecting the payload symbols contained in $\{\dot{\mathbf{x}}_n\}$ is a well-defined ISI equalization problem, and it can be tackled by employing, for example, the RELAX-BLAST algorithm.

4.2.3 Alamouti Diversity Scheme

As an alternative to the spatial multiplexing scheme including RELAX-BLAST, we can also use the space-time coding scheme such as the Alamouti diversity technique to facilitate the symbol detection task for reduced bit error rate (BER). The Alamouti diversity scheme is practically attractive since it fully exploits the transmit diversity and generally allows for very efficient equalization [2, 44]. For completeness, the Alamouti diversity scheme is briefly reviewed in this section.

We aim to transmit $2L$ payload symbols using a system equipped with $N = 2$ transmitters and $M = 1$ receiver over ISI channels. The $2L$ symbols are divided into two segments, say $\mathbf{a}$ and $\mathbf{b}$, each of length L. Per the suggestions presented in [44], the structure of the two transmitted sequences is shown in Figure 4-1. Each transmitter sends the signal in two separate bursts. In Figure 4-1, for example, the first transmitter sends $\mathbf{a}$ and $-\mathbf{b}^\dagger$ during the first and second bursts, respectively. $\mathbf{b}^\dagger$ denotes a conjugated time-reversed version of $\mathbf{b}$ (e.g., if $\mathbf{b} = [b(1), \ldots, b(L)]^T$, then $\mathbf{b}^\dagger = [b^*(L), \ldots, b^*(1)]^T$). The second transmitter transmits $\mathbf{b}$ and $\mathbf{a}^\dagger$ during the first and second bursts, respectively. Between the two bursts lies a gap. Typically, the gap length is larger than the channel tap number R to ensure that the interference from the first burst does not extend to the second burst.

Define $\bar{\mathbf{H}}_{n,1}$ and $\bar{\mathbf{H}}_{n,1}^{\dagger} \in \mathbb{C}^{(R+L-1)\times L}$, respectively, as:

$$\bar{\mathbf{H}}_{n,1} = \begin{bmatrix} \hat{h}_{n,1}(1) & & \mathbf{0} \\ \vdots & \ddots & \\ \hat{h}_{n,1}(R) & & \hat{h}_{n,1}(1) \\ & \ddots & \vdots \\ \mathbf{0} & & \hat{h}_{n,1}(R) \end{bmatrix} \quad \text{and} \quad \bar{\mathbf{H}}_{n,1}^{\dagger} = \begin{bmatrix} \hat{h}_{n,1}^{*}(R) & & \mathbf{0} \\ \vdots & \ddots & \\ \hat{h}_{n,1}^{*}(1) & & \hat{h}_{n,1}^{*}(R) \\ & \ddots & \vdots \\ \mathbf{0} & & \hat{h}_{n,1}^{*}(1) \end{bmatrix}, \quad n = 1, 2. \tag{4–17}$$

Once the CIR estimates $\{\hat{\mathbf{h}}_{n,1}\}_{n=1}^{2}$ and the phase compensated measurement vector $\tilde{\mathbf{y}}_1$ are available, $\tilde{\mathbf{y}}_{1,l}$, a portion of $\tilde{\mathbf{y}}_1$ corresponding to the first burst (Figure 4-1), can be expressed as :

$$\tilde{\mathbf{y}}_{1,l} = \bar{\mathbf{H}}_{1,1}\mathbf{a} + \bar{\mathbf{H}}_{2,1}\mathbf{b} + \tilde{\mathbf{e}}_{1,l}. \tag{4–18}$$

Similarly, $\tilde{\mathbf{y}}_{1,r}$, a portion of $\tilde{\mathbf{y}}_1$ corresponding to the second burst, is given by:

$$\tilde{\mathbf{y}}_{1,r} = \bar{\mathbf{H}}_{1,1}\left(-\mathbf{b}^{\dagger}\right) + \bar{\mathbf{H}}_{2,1}\mathbf{a}^{\dagger} + \tilde{\mathbf{e}}_{1,r}. \tag{4–19}$$

Note that (4–18) and (4–19) implicitly assume that the assumed ISI channel remains constant over one block time; see Figure 4-1.

Combining $\tilde{\mathbf{y}}_{1,l}$ and $\tilde{\mathbf{y}}_{1,r}^{\dagger}$ follows

$$\begin{bmatrix} \tilde{\mathbf{y}}_{1,l} \\ \tilde{\mathbf{y}}_{1,r}^{\dagger} \end{bmatrix} = \begin{bmatrix} \bar{\mathbf{H}}_{1,1} & \bar{\mathbf{H}}_{2,1} \\ \bar{\mathbf{H}}_{2,1}^{\dagger} & -\bar{\mathbf{H}}_{1,1}^{\dagger} \end{bmatrix} \begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix} + \begin{bmatrix} \tilde{\mathbf{e}}_{1,l} \\ \tilde{\mathbf{e}}_{1,r}^{\dagger} \end{bmatrix} = \bar{\mathbf{H}} \begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix} + \begin{bmatrix} \tilde{\mathbf{e}}_{1,l} \\ \tilde{\mathbf{e}}_{1,r}^{\dagger} \end{bmatrix}. \tag{4–20}$$

It is easy to verify that the matrix product $\bar{\mathbf{H}}^{H}\bar{\mathbf{H}} \in \mathbb{C}^{2L\times 2L}$ possesses a block diagonal structure: the entries of the two $L \times L$ matrices on the off-diagonal position are all zero, whereas the entries of the two $L \times L$ matrices on the diagonal position are generally nonzero. This suggests that by making use of the Alamouti diversity scheme, matched filter (by multiplying $\bar{\mathbf{H}}^{H}$ on both sides of (4–20)) decomposes the original detection problem of dimension $2L$ into two subproblems, each of dimension L (i.e., the detection

of $\mathbf{a}$ and $\mathbf{b}$ is performed seperately) [44]. Note that for each subproblem, both of the payload sequences $\mathbf{a}$ and $\mathbf{b}$ still suffer from ISI, making equalization necessary [44].

Ideally, if the columns in $\bar{\mathbf{H}}$ in (4–20) are pairwise orthogonal (i.e., $\bar{\mathbf{H}}^H\bar{\mathbf{H}}$ becomes diagonal), then the matched filter could decouple the original detection problem into a bank of $2L$ independent scalar problems. Moreover, under the Gaussian noise assumption, the so-obtained matched filter result is equivalent to the maximum likelihood result. It has been demonstrated in [1] that such pairwise orthogonality property can be achieved in the frequency domain. To see this, it is suggestive to first review the diagonalization property of a circulant matrix. It is well-known that a $K \times K$ complex-valued circulant matrix $\mathbf{C}$ can be diagonalized as $\mathbf{C} = \mathbf{F}^H\mathbf{\Gamma}\mathbf{F}$ [71]. Here, $\mathbf{F}$ is the $K \times K$ FFT matrix and $\mathbf{\Gamma}$ holds the K eigenvalues of $\mathbf{C}$ on its diagonal [71]. Furthermore, these eigenvalues can be efficiently computed by applying FFT to the first column of $\mathbf{C}$. Note that both $\{\bar{\mathbf{H}}_{i,1}\}_{i=1}^2$ and $\{\bar{\mathbf{H}}_{i,1}^\dagger\}_{i=1}^2$ in (4–20) possess the Toeplitz structure, which can be expanded to circulant matrices by appending the appropriate $(R+L-1) \times (R-1)$ matrix to the right hand side of the respective matrix. That is, $\bar{\mathbf{C}}_{i,1}$ ($\bar{\mathbf{C}}_{i,1}^\dagger$) can be constructed by cyclically shifting the last column of $\bar{\mathbf{H}}_{i,1}$ ($\bar{\mathbf{H}}_{i,1}^\dagger$) for $R-1$ times to the bottom for $i = 1, 2$. Then, (4–20) can be rewritten as

$$\begin{bmatrix} \tilde{\mathbf{y}}_{1,l} \\ \tilde{\mathbf{y}}_{1,r}^\dagger \end{bmatrix} = \begin{bmatrix} \bar{\mathbf{C}}_{1,1} & \bar{\mathbf{C}}_{2,1} \\ \bar{\mathbf{C}}_{2,1}^\dagger & -\bar{\mathbf{C}}_{1,1}^\dagger \end{bmatrix} \begin{bmatrix} \check{\mathbf{a}} \\ \check{\mathbf{b}} \end{bmatrix} + \begin{bmatrix} \tilde{\mathbf{e}}_{1,l} \\ \tilde{\mathbf{e}}_{1,r}^\dagger \end{bmatrix} \tag{4–21}$$

$$= \begin{bmatrix} \mathbf{F}^H\bar{\mathbf{\Gamma}}_1\mathbf{F} & \mathbf{F}^H\bar{\mathbf{\Gamma}}_2\mathbf{F} \\ \mathbf{F}^H\bar{\mathbf{\Gamma}}_3\mathbf{F} & -\mathbf{F}^H\bar{\mathbf{\Gamma}}_4\mathbf{F} \end{bmatrix} \begin{bmatrix} \check{\mathbf{a}} \\ \check{\mathbf{b}} \end{bmatrix} + \begin{bmatrix} \tilde{\mathbf{e}}_{1,l} \\ \tilde{\mathbf{e}}_{1,r}^\dagger \end{bmatrix}. \tag{4–22}$$

In (4–21), $\check{\mathbf{a}}$ ($\check{\mathbf{b}}$) is obtained by padding $R-1$ zeros at the end of $\mathbf{a}$ ($\mathbf{b}$), and (4–22) follows directly from the diagonalization property of the circulant matrices.

Equation (4–22) can be rewritten as:

$$\begin{bmatrix} \mathbf{F}\tilde{\mathbf{y}}_{1,l} \\ \mathbf{F}\tilde{\mathbf{y}}_{1,r}^\dagger \end{bmatrix} = \begin{bmatrix} \bar{\mathbf{\Gamma}}_1 & \bar{\mathbf{\Gamma}}_2 \\ \bar{\mathbf{\Gamma}}_3 & -\bar{\mathbf{\Gamma}}_4 \end{bmatrix} \begin{bmatrix} \mathbf{F}\check{\mathbf{a}} \\ \mathbf{F}\check{\mathbf{b}} \end{bmatrix} + \begin{bmatrix} \mathbf{F}\tilde{\mathbf{e}}_{1,l} \\ \mathbf{F}\tilde{\mathbf{e}}_{1,r}^\dagger \end{bmatrix}. \tag{4–23}$$

Note that multiplying $\mathbf{F}$ on the left side of a vector is an FFT operation. Therefore, (4–23) in effect converts our view point to the frequency domain. We express (4–23) in a more compact form:

$$\mathbf{Y} = \bar{\mathbf{\Gamma}}\mathbf{D} + \mathbf{E}. \tag{4–24}$$

In the present paper, this is the only time that we use boldface uppercase letters $\mathbf{Y}$, $\mathbf{D}$ and $\mathbf{E}$ to represent, respectively, the frequency representations of the measurement vector, the estimate targets and noise terms. Further, $\mathbf{E} \sim \mathcal{CN}(0, \eta\mathbf{I})$ follows from the fact that both FFT and conjugate time-reversion operation preserve the statistical property of the noise terms $\tilde{\mathbf{e}}_{1,l}$ and $\tilde{\mathbf{e}}_{1,r}$. The key advantage of the frequency view point is that now the columns of $\bar{\mathbf{\Gamma}}$ are pairwise orthogonal (i.e., $\bar{\mathbf{\Gamma}}^H\bar{\mathbf{\Gamma}}$ is diagonal), in contrast to (4–20) where $\bar{\mathbf{H}}^H\bar{\mathbf{H}}$ exhibits a block diagonal structure. Therefore, in our example, the matched filter is employed to obtain $\hat{\mathbf{D}}$. Once $\hat{\mathbf{D}}$ is available, the estimate of $\check{\mathbf{a}}$ and $\check{\mathbf{b}}$ can be obtained via inverse FFT (IFFT). Finally, the estimate of $\mathbf{a}$ ($\mathbf{b}$) in (4–20) can be obtained from the estimate of $\check{\mathbf{a}}$ ($\check{\mathbf{b}}$) by straightforward re-indexing: the estimate of $\mathbf{a}$ ($\mathbf{b}$) can be obtained from the first L elements of the estimate of $\check{\mathbf{a}}$ ($\check{\mathbf{b}}$).

4.3 Numerical and Experimental Results

4.3.1 Simulation of Channel Estimation Performance

We compare the channel estimation performance using GoSLIM and a method that estimates Doppler frequencies and CIRs separately. The latter scheme, referred to as two-step method, is implemented as follows. First, the Doppler frequencies are estimated by solving the optimization problem:

$$\hat{f} = \arg\max_{\tilde{f}} \left\| \left(\tilde{\mathbf{\Lambda}}\mathbf{X}\right)^H \mathbf{y} \right\|^2, \tag{4–25}$$

where $\tilde{\mathbf{\Lambda}}$ is modeled similarly to (4–4) as:

$$\tilde{\mathbf{\Lambda}} = \mathrm{diag}\left(\left[1, e^{-2j\pi\tilde{f}T_s}, \ \ldots \ , e^{-2j\pi\tilde{f}T_s(P-1)}\right]\right), \tag{4–26}$$

with $\tilde{f}$ being the assumed Doppler frequency. Once $\hat{f}$ is available, we compensate for the Doppler effects on the received measurements, and then estimate the CIRs via the linear minimum mean-squared error (LMMSE) filtering:

$$\hat{\mathbf{h}} = \left(\mathbf{X}^H\mathbf{X} + \eta\mathbf{I}\right)^{-1}\mathbf{X}^H\tilde{\mathbf{y}}, \tag{4–27}$$

where $\tilde{\mathbf{y}} = \hat{\mathbf{\Lambda}}^H\mathbf{y}$ represents the measurement vector after Doppler compensation and $\hat{\mathbf{\Lambda}}$ still follows (4–26) but with $\tilde{f}$ replaced by $\hat{f}$.

Consider a UAC system equipped with $N = 2$ transmitters and $M = 1$ receiver. The modulus of the simulated sparse CIRs corresponding to the two transmitters are shown in Figure 4-2 with $R = 20$ taps. In our simulation, we set the Doppler frequency and the symbol period as $f = 1$ Hz and $T_s = 0.125$ ms, respectively, and assume that both the simulated CIRs and f are constant. Two different types of training sequences are compared: one is the shifted PeCAN sequence, which could be used in the training-directed channel estimation, and the other is the random QPSK sequence to resemble the decision-directed channel estimation scenario. Both sequences have a length of $P = d_y = 256$ symbols. The PeCAN sequences are used with $L_{\mathrm{CP}} = R - 1 = 19$ cyclic prefix symbols, while for QPSK sequences, the preceding 19 symbols are randomly generated QPSK symbols. Given the CIR truths, training sequences, f and T_s, the received measurements are constructed as in (4–5) with $\mathbf{e} \sim \mathcal{CN}(\mathbf{0}, \eta\mathbf{I})$. Note that for the MMSE CIR estimator in (4–27), the true noise variance η is given as a priori knowledge (in practice, however, η needs to be estimated).

When the shifted PeCAN sequences are employed, the mean-squared errors (MSEs) of the CIR estimates and the Doppler frequency estimates obtained by GoSLIM and the two-step method versus the noise power η are shown in Figures 4-3A and 4-3C, respectively, and those for QPSK sequences are shown in Figures 4-3B and 4-3D, respectively. Each point here is averaged over 100 Monte-Carlo trials. The noise pattern varies independently for each trial. By comparing Figures 4-3C with 4-3D, one observes

that the Doppler frequency estimator (4–25) is quite sensitive to the characteristics of the training sequences used. In particular, the MSE of the frequency estimate for QPSK sequences is much worse than that for the shifted PeCAN sequences. To verify this observation, we consider a data model in the absence of noise (i.e., the noise term $\mathbf{e}$ is dropped in (4–6)):

$$\mathbf{y} = \mathbf{\Lambda X h}. \tag{4–28}$$

We first construct $\tilde{\mathbf{\Lambda}}$ as in (4–26), and then left multiply $\left(\tilde{\mathbf{\Lambda}}\mathbf{X}\right)^H$ to both sides of (4–28):

$$\left(\tilde{\mathbf{\Lambda}}\mathbf{X}\right)^H \mathbf{y} = \mathbf{X}^H \tilde{\mathbf{\Lambda}}^H \mathbf{\Lambda X h}. \tag{4–29}$$

When the shifted PeCAN sequences are employed, $\mathbf{X}^H\mathbf{X}$ essentially equals $d_y\mathbf{I}$ [26], and $\tilde{\mathbf{\Lambda}} = \mathbf{\Lambda}$ is the solution to (4–25), leading to a perfect frequency estimate, i.e., $\hat{f} = f$. We start the proof by plugging (4–28) into the cost function in (4–25), which gives:

$$\left\|(\tilde{\mathbf{\Lambda}}\mathbf{X})^H\mathbf{y}\right\|^2 = \left\|\mathbf{X}^H\tilde{\mathbf{\Lambda}}^H\mathbf{\Lambda X h}\right\|^2 = \left\|\mathbf{X}^H\bar{\mathbf{\Lambda}}\mathbf{X h}\right\|^2,$$

where

$$\bar{\mathbf{\Lambda}} \triangleq \tilde{\mathbf{\Lambda}}^H\mathbf{\Lambda} = \mathrm{diag}\left(\left[1, e^{-2j\pi\Delta f T_s}, \ \ldots \ , e^{-2j\pi\Delta f T_s(d_y-1)}\right]\right), \tag{4–30}$$

with $\Delta f \triangleq f - \tilde{f}$. It is easy to see that $\bar{\mathbf{\Lambda}}$ is unitary. For notational simplicity, we henceforth drop the constant d_y and rewrite $\mathbf{X}^H\mathbf{X} = \mathbf{I}_{NR\times NR}$, i.e., $\mathbf{X}$ becomes semi-unitary. (This can be achieved by scaling each element in $\mathbf{X}$ by $\frac{1}{\sqrt{d_y}}$.) We append the appropriate $d_y \times (d_y - NR)$ matrix to the right-hand side of $\mathbf{X} \in \mathbb{C}^{d_y\times NR}$ and construct a unitary matrix $\left[\mathbf{X}\ \tilde{\mathbf{X}}\right] \in \mathbb{C}^{d_y\times d_y}$. Then:

$$\left[\mathbf{X}\ \tilde{\mathbf{X}}\right]^H \bar{\mathbf{\Lambda}}^H \left[\mathbf{X}\ \tilde{\mathbf{X}}\right] \left[\mathbf{X}\ \tilde{\mathbf{X}}\right]^H \bar{\mathbf{\Lambda}} \left[\mathbf{X}\ \tilde{\mathbf{X}}\right] = \mathbf{I}_{d_y\times d_y}. \tag{4–31}$$

The left-hand side of Equation (4–31) can be partitioned as follows:

$$\begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{bmatrix} = \mathbf{I}_{d_y\times d_y},$$

where $\mathbf{A}_{11} \in \mathbb{C}^{NR\times NR}$, $\mathbf{A}_{12} \in \mathbb{C}^{NR\times(d_y-NR)}$, $\mathbf{A}_{21} \in \mathbb{C}^{(d_y-NR)\times NR}$ and $\mathbf{A}_{22} \in \mathbb{C}^{(d_y-NR)\times(d_y-NR)}$. Confining our focus to the submatrix $\mathbf{A}_{11}$, we get:

$$\mathbf{A}_{11} = \mathbf{X}^H\bar{\mathbf{\Lambda}}^H\mathbf{X}\mathbf{X}^H\bar{\mathbf{\Lambda}}\mathbf{X} + \mathbf{X}^H\bar{\mathbf{\Lambda}}^H\tilde{\mathbf{X}}\tilde{\mathbf{X}}^H\bar{\mathbf{\Lambda}}\mathbf{X} = \mathbf{I}_{NR\times NR}.$$

Thus

$$\mathbf{h}^H\mathbf{A}_{11}\mathbf{h} = \mathbf{h}^H\mathbf{X}^H\bar{\mathbf{\Lambda}}^H\mathbf{X}\mathbf{X}^H\bar{\mathbf{\Lambda}}\mathbf{X}\mathbf{h} + \mathbf{h}^H\mathbf{X}^H\bar{\mathbf{\Lambda}}^H\tilde{\mathbf{X}}\tilde{\mathbf{X}}^H\bar{\mathbf{\Lambda}}\mathbf{X}\mathbf{h} = \mathbf{h}^H\mathbf{h}.$$

Since $\mathbf{X}^H\bar{\mathbf{\Lambda}}^H\tilde{\mathbf{X}}\tilde{\mathbf{X}}^H\bar{\mathbf{\Lambda}}\mathbf{X}$ is positive semi-definite, we have

$$\mathbf{h}^H\mathbf{X}^H\bar{\mathbf{\Lambda}}^H\tilde{\mathbf{X}}\tilde{\mathbf{X}}^H\bar{\mathbf{\Lambda}}\mathbf{X}\mathbf{h} \geq 0.$$

As a consequence,

$$\mathbf{h}^H\mathbf{X}^H\bar{\mathbf{\Lambda}}^H\mathbf{X}\mathbf{X}^H\bar{\mathbf{\Lambda}}\mathbf{X}\mathbf{h} \leq \mathbf{h}^H\mathbf{h},$$

which is equivalent to

$$\left\|\mathbf{X}^H\bar{\mathbf{\Lambda}}\mathbf{X}\mathbf{h}\right\|^2 \leq \|\mathbf{h}\|^2.$$

The upper bound is achieved for $\bar{\mathbf{\Lambda}} = \mathbf{I}$, i.e., $\Delta f = 0$ (Equation (4–30)). Consequently, $\hat{f} = f$ is the solution to (4–25).

As a consequence, one can expect that as η decreases, the MSE curve in Figure 4-3C will decrease and become zero as η goes to zero. In contrast, when applying the QPSK sequences (i.e., $\mathbf{X}^H\mathbf{X} \neq d_y\mathbf{I}$), $\tilde{\mathbf{\Lambda}} = \mathbf{\Lambda}$ is no longer the solution to (4–25), which results in a biased frequency estimate. This is in line with the observation that the MSE curve in Figure 4-3D is limited to a certain level. Such biased Doppler frequency estimates will further degrade the subsequent MMSE-based CIR estimate, as observed by comparing Figures 4-3A and 4-3B.

GoSLIM, on the other hand, is robust against the types of the sequences used, which is a desired property for a channel estimation algorithm. We can see from Figure 4-3C that the MSE of the frequency estimate given by GoSLIM is slightly better than that obtained using (4–25) due to its joint estimation mechanism. GoSLIM gives consistently

better CIR estimate than its MMSE counterpart (Figures 4-3A and 4-3B) even if the latter has the noise power η known a priori. This is mainly because GoSLIM addresses sparsity while MMSE does not. In sum, the two-step method is inferior than GoSLIM, especially when the training sequences do not possess good correlation properties. Due to this reason, we will use GoSLIM to analyze the experimental data in the following.

4.3.2 WHOI09 In-Water Experimentation Results

4.3.2.1 Experiment specifics

The WHOI09 in-water experiment was conducted in December 2009. The 4 transmit transducers, with source spacing up to 1 m, were suspended from a vessel heaving in a 14 m mid-depth water column. Two separate 4-hydrophone arrays were deployed approximately 2 km and 1 km away from the source array and they are referred to, respectively, as the RB1 and RB2 receiving arrays. Both of the RB1 and RB2 receiving arrays had 0.21 m spacing between adjacent hydrophones, and they were mounted on anchored buoys in a mid-water column during the course of data collection. For more details about the experiment, we refer the readers to [89]. The Doppler effects were mainly due to the relative motion between the transmitters and receivers. The carrier frequency, the sampling frequency and the symbol rate employed in the WHOI09 experiment were 30 kHz, 200 kHz and 8 kHz, respectively. The Alamouti schemes were thoroughly tested in the WHOI09 experiment. In particular, we explored the duality between the UAC systems equipped with 1 transmitter and 2 receivers (1Tx$-$2Rx) and 2 transmitters and 1 receiver (2Tx$-$1Rx), both attaining a coded (uncoded) data rate of 3.5 kbps (7 kbps). We also show at the cost of increased computational complexity at the receiver side, 2 pairs of Alamouti codes could also be transmitted simultaneously to double the data rate using a receiver array with multiple elements. In WHOI09, the transmitted signal was recorded by both RB1 and RB2 and each of the data packages we design was transmitted 3 times. Consequently, a total of 6 epochs were available for our processing. In what follows, an epoch is named, for example, “195600-RB1”, which

refers to the measurements acquired by RB1 in response to the signal transmitted at time “195600”.

To examine the channel conditions, GoSLIM is employed to estimate the underlying CIRs and Doppler frequencies by making use of the shifted PeCAN training sequences periodically allocated over the transmitted sequence. Figure 4-4 shows the CIR evolutions between the active transmitter and the 2nd receive hydrophone for each epoch. In addition, Figure 4-5 plots the evolutions of the Doppler frequencies for each hydrophone. One can see that the position of the principal arrival and the surface-interactive paths shown in Figure 4-4 are slowly shifting leftwards with time due to temporal compression: over the 9 s period of transmission (during which a total of 71.1 k symbols were transmitted), the signal is compressed by 4 symbols at most (with Figure 4-4(a) being the worst case). As a consequence, the effects of time stretching are very limited and can be neglected during each block of 0.14 s.

4.3.2.2 Performance of the Alamouti coding scheme

To investigate the performance of the 2Tx$-$1Rx Alamouti scheme, the structure of the transmitted sequences is shown in Figure 4-6. Two synchronized transmitters were activated, and each transmitter sent four data packets in succession. Each data packet can be further divided into 16 blocks followed by a gap of length 500. The presence of the gap ensures the elimination of inter-packet interferences. The construction of each block pair across the two active transmitters is quite similar to that in Figure 4-1, except that the gap between the two payload segments in Figure 4-1 is now replaced by a training sequence formed by a shifted PeCAN sequence with $P = 512$ symbols and $L_{\mathrm{CP}} = 99$ cyclic prefix symbols. In our design, each payload segment (e.g., $\mathbf{a}$ or $\mathbf{b}$ in Figure 4-6) contains $L = 250$ QPSK symbols and has undergone channel coding. Taking segment $\mathbf{a}$ for example, it is generated, as shown in Figure 4-6, by feeding 250 source bits into a $1/2$ rate convolutional encoder with generator polynomials (1 0 0 1 1) and (1 1 0 1 1) followed by a random interleaver and QPSK modulation using Gray code

mapping. Segment $\mathbf{b}$, as well as the segments in other blocks, is similarly generated but with different source bits and a different random interleaver. Note that inner-block interferences caused by the training sequences can be easily removed by subtracting their contributions out from the measurements after the channel estimation has been done, while inter-block interferences caused by payload sequences can be mitigated by allocating guard intervals between adjacent blocks, as suggested in [44]. The Alamouti structure shown in Figure 4-6 leads to an uncoded (coded) data rate of approximately 7 kbps (3.5 kbps).

Since each payload block contains its own training sequence (Figure 4-6), symbol detection can be performed on a block-by-block basis: we first conduct training-directed channel estimation, and then detect the 500 QPSK payload symbols within the block of current interest per the discussions in Section 4.2.3. As a consequence, decision-directed channel estimation is not needed. In our analysis, the channel tap number is fixed at $R = 30$ for all epochs. The empirical BER results at different receive hydrophones, by averaging over 192 k uncoded bits and 96 k coded bits (recall that the transmitted sequence shown in Figure 4-6 carries 32 k QPSK payload symbols, i.e., 64 k uncoded bits or 32 k coded bits, and we have 3 epochs for each receiving array), are summarized in Table 4-1. One observes from Table 4-1 that the 2nd hydrophone of the RB1 array yields significantly higher BERs than others. This can be explained by looking at Figure 4-4, which shows the comparison of the estimated CIRs for the 2nd hydrophone of the RB1 and RB2 arrays. Note that the channel amplitude at the 2nd hydrophone of RB1 is lower than that of RB2 by almost an order of magnitude. It is worth pointing out that when analyzing the 2Tx$-$1Tx Alamouti scheme, we could also extend the GoSLIM algorithm to allow each transmitter to have its own Doppler frequency, instead of a common one as assumed by GoSLIM. This extension, however, did not result in visible performance improvement over the original GoSLIM, and it suffers from a higher computational complexity compared to the latter.

Table 4-1. BER performance of GoSLIM coupled with Alamouti diversity scheme for 2Tx-1Rx systems.

	RB1		RB2	
	uncoded BER	coded BER	uncoded BER	coded BER
Rx1	2.7×10^{-4}	0	1.9×10^{-4}	0
Rx2	1.8×10^{-2}	2.1×10^{-3}	1.3×10^{-4}	0
Rx3	1.6×10^{-4}	0	6.2×10^{-5}	0
Rx4	3.2×10^{-4}	0	1.1×10^{-4}	0

Table 4-2. BER performance of using GoSLIM for 1Tx-2Rx systems.

	RB1		RB2	
indices of the Rxs	uncoded BER	coded BER	uncoded BER	coded BER
$\{1, 2\}$	3.0×10^{-3}	2.3×10^{-3}	0	0
$\{1, 3\}$	5.2×10^{-6}	0	0	0
$\{1, 4\}$	5.2×10^{-6}	0	0	0
$\{2, 3\}$	2.7×10^{-3}	8.9×10^{-4}	1.6×10^{-5}	0
$\{2, 4\}$	6.6×10^{-3}	3.5×10^{-3}	2.1×10^{-5}	0
$\{3, 4\}$	4.0×10^{-4}	1.0×10^{-5}	5.2×10^{-6}	0

In addition to the conventional 2Tx$-$1Rx Alamouti coding technique, we also tried the 1Tx$-$2Rx scheme by activating only Tx1 in Figure 4-6. (This single-input sequence is used to obtain Figures 4-4 and 4-5.) Note that the data rate of the 1Tx$-$2Rx system is the same as that of its 2Tx$-$1Rx counterpart. The uncoded and coded BER results obtained by GoSLIM with the 1Tx$-$2Rx system configuration are summarized in Table 4-2. Once again, the BER results, both uncoded and coded, become poor whenever the second hydrophone of the RB1 array is used. Once again, MF yields much higher BER than GoSLIM. Comparing Tables 4-1 with 4-2, one observes that the BER results of the 2Tx$-$1Rx system is worse than that of the 1Tx$-$2Rx counterpart when using GoSLIM. This could be due to the fact that the former system has more unknown parameters (per receiver).

Table 4-3. BER performance of using GoSLIM with LMMSE for transmitting 2 pairs of Alamouti codes.

	RB1		RB2	
	uncoded BER	coded BER	uncoded BER	coded BER
the 1st pair	2.0×10^{-4}	0	1.0×10^{-4}	0
the 2nd pair	1.5×10^{-4}	0	2.3×10^{-3}	0

4.3.2.3 Performance of transmitting 2 pairs of Alamouti codes

As previously remarked, 2Tx−1Rx and 1Tx−2Rx system configurations result in the same data rate (i.e., 7 kbps uncoded data rate and 3.5 kbps coded). This data rate could be doubled if we transmit 2 pairs of Alamouti sequences by activating all of the 4 transmitters: with Tx1 and Tx2 forming one pair, and Tx3 and Tx4 forming another. The construction of each pair still follows Figure 4-6 with each packet consisting of 8 blocks. As a consequence, in each epoch, the 1st transmitter pair carries 16 k QPSK payload symbols, while the 2nd pair carries another 16 k symbols. Note that the price we pay for this doubled data rate is the increased reception complexity. For the 2-pair Alamouti configurations, the orthogonality property (Section 4.2.3) still holds within each transmitter pair in the frequency domain, which is ensured by the structure of the transmitted signals (Figure 4-1). The orthogonality property, however, no longer holds across the two transmitter pairs. Consequently, the detection problem at hand cannot be decoupled into a bank of scaler problems using the matched filter. To deal with this problem, the LMMSE based RELAX-BLAST scheme is employed for symbol detection. To take advantage of the spacial diversity, we use the measurements from the 1st, 3rd and 4th receive hydrophones for symbol detection (the 2nd hydrophones are avoided purposely due to the defective performance of the 2nd hydrophone of RB1, as remarked previously). The resulting BER results, obtained by averaging over 96 k uncoded bits and 48 k coded bits, are summarized in Table 4-3.

4.3.3 ACOMM10 In-Water Experimentation Results

4.3.3.1 Experiment specifics

The ACOMM10 experiment took place in July 2010 in the Mid-Atlantic Bight on the continental shelf off the coast of New Jersey in an area with water depth of 78 m. The transmitter array consisted of 12 transducers with 0.8 m spacing between adjacent elements. In our design, only 4 transducers were activated, i.e., $N = 4$. The receiving array was composed of 8 hydrophones, i.e., $M = 8$. The spacing between the adjacent hydrophones was 2.06 m except the first element, which was spaced 4 m above the second element. Both the transmit array and the receiving array were mounted on anchored buoys and were deployed approximately 3 km away from each other. The carrier frequency, the sampling frequency and the symbol rate employed in the ACOMM10 experiment were 20 kHz, 80 kHz and 4 kHz, respectively.

The BLAST data package of 20 s in duration was transmitted multiple times in the ACOMM10 experiment and was recorded by the receiving array. A total of 89 epochs were available and they are referred to as "MIMO01"–"MIMO89", respectively. Figures 4-7 and 4-8 show, respectively, the evolutions of CIR and Doppler frequencies estimated by GoSLIM. To obtain these figures, we treat the 4 simultaneously transmitted sequences as perfectly known at the receiver side and we use a tracking length of $L = 450$.

4.3.3.2 Performance of the MIMO BLAST scheme

Figure 4-9 shows the source information contained in the transmitted package. Each package consists of 7 packets. The first 4 packets convey 4 grayscale Gator mascots and the last 3 packets combined form a colored mascot. The RGB components of the colored image are transmitted in the 5th, 6th and 7th packets, respectively. Guard intervals between adjacent packets are used to prevent inter-packet interference. Each pixel of the grayscale image is represented by 5 bits, corresponding to 32 different intensities (e.g., pure white and pure dark pixels are represented by 11111 and 00000,

respectively). The 64-pixel by 100-pixel grayscale mascot image, as a consequence, is represented by a total of 32 k source bits. (Accordingly, a colored mascot image is represented by 96 k bits.) The contrast of the grayscale image, as well as the hue of the colored image, has been carefully adjusted so that the image carries approximately equal numbers of 1's and 0's.

We herein elaborate how to generate a packet from the grayscale Gator mascot image (the packet generation for each of the RGB components of the colored image follows the same procedure). Specifically, the 32 k source bits are first interleaved so that the bits feeding into the convolutional encoder module have an equal chance of being 0 or 1; see Figure 4-10. The so-obtained 32 k interleaved source bits are then divided into 32 groups, each contains 1 k bits. The ith group ($i = 1, \ldots, 32$) will be used to construct the ith payload symbol block across the 4 transmitted sequences. To see this, Figure 4-10 illustrates a scenario with $i = 1$ (the methodology can be mapped to the cases with $i = 2, \ldots, 32$ in a straightforward manner). We first feed the 1 k bits contained in the 1st group into a $1/2$ rate convolutional encoder with generator polynomials (1 0 0 1 1) and (1 1 0 1 1). The 2 k encoded bits are then passed to another random interleaver, followed by QPSK modulation using Gray code mapping. The resulting 1 k QPSK payload symbols are finally demultiplexed into the 4 payload blocks (each contains 250 QPSK symbols) across the 4 transmitters in a round-robin fashion (this is where the "Demultiplexer" module in Figure 4-10 comes into play). More specifically, in our design, the symbols with index $\{n + 4a\}_{a=0}^{249}$ form the payload block at the nth transmitter ($n \in \{1, 2, 3, 4\}$). The shifted PeCAN training sequences with length $P = 512$, in conjunction with $L_{\text{CP}} = 99$ cyclic prefix symbols, form the training section, which is located between the 16th and 17th payload blocks. This MIMO UAC design leads to a net coded data rate of 15 kbps.

By transmitting $N = 4$ sequences simultaneously and incorporating the measurements acquired from all off the $M = 8$ receiver elements for analysis, we established a 4×8

MIMO UAC system. Unlike the Alamouti sequences adopted in the WHOI09 experiment where the training-directed channel estimation can be conducted periodically, only one training sequence is used for the BLAST data in ACOMM10 (Figure 4-10). Therefore, the decision-directed channel estimation becomes indispensable when analyzing the ACOMM10 data. The channel tap number and the tracking length are fixed, respectively, at $R = 55$ and $L = 450$ for all of the 89 epochs. The channel tracking starts with training-directed channel estimation using GoSLIM. Then we perform phase compensation separately at each receiving hydrophone as done in (4–16) before proceeding to employ RELAX-BLAST to detect the first 250 payload symbols contained in the 17th payload block for each transmitted sequence. Next, the channels are updated in the decision-directed mode using 450 symbols (containing the previously detected payload symbols, as well as a portion of the training sequence as well). With the updated CIRs and Doppler frequencies, after phase compensation, the subsequent 250 payload symbols contained in the 18th block are detected using RELAX-BLAST. This process continues until all of the 16 payload blocks to the right-hand side of the training sequences are detected. This same tracking scheme can be applied in a reverse manner to the detection of the 16 payload blocks ahead of the training sequences.

We deem a packet to be successfully detected if its coded BER is less than 0.1. By adopting the aforementioned reception scheme, we have succeeded in tracking the entire 32 payload blocks for 594 out of the 623 packets (recall that we have 89 epochs and each consists of 7 packets). A coded BER of 5.1×10^{-3} is achieved after averaging over the 1.9×10^{7} source bits processed. Among the 594 successful packets, we selected some packets to demonstrate the impact of coded BER on the quality of the recovered mascot images. Figures 4-11A and 4-11D are from epoch “MIMO08”, corresponding to BERs of 0 and 5.0×10^{-5}, respectively. These remarkable BER results translate into almost perfect image recovery. Figures 4-11B and 4-11E are from epoch “MIMO03”, corresponding to BERs of 2.9×10^{-3} and 5.0×10^{-3}, respectively. One

observes that the mascot details are still well preserved despite of the presence of the sparse noisy dots due to the moderate amount of bit errors. Figures 4-11C and 4-11F are from epoch "MIMO27", corresponding to BERs of 6.8×10^{-3} and 3.2×10^{-2}, respectively. These BER results lead to further degraded reconstructed images.

For comparison purposes, we proceed to assess the detection performance of SLIM [46], i.e., with an ISI channel model instead of a double spreading channel model, by analyzing the data packet that leads to perfect recovery in Figure 4-11A using GoSLIM. The same reception scheme is repeated (except that the phase compensation stage is no longer needed since SLIM assumes $f = 0$) and the recovered mascot image using SLIM is shown in Figure 4-12. The image becomes hard to identify since more detection errors occur (final coded BER is 2.2×10^{-1}). This suggests that in our examples, it becomes indispensable to take the Doppler effects into consideration for achieving reliable UAC.

Finally, we remark that one packet of grayscale mascot image was also transmitted in the WHOI09 experiment using $N = 4$ transducers and $M = 4$ hydrophones at a coded data rate of 30 kbps. The empirical coded BERs are 1.7×10^{-3}, 1.3×10^{-3} and 4.7×10^{-4} over the entire 32 k source bits for epochs "195600-RB2", "195730-RB2" and "195860-RB2", respectively. The measurements obtained using the RB1 array, however, cannot be used to reconstruct reasonable mascots due to the problem suffered by the second hydrophone of RB1.

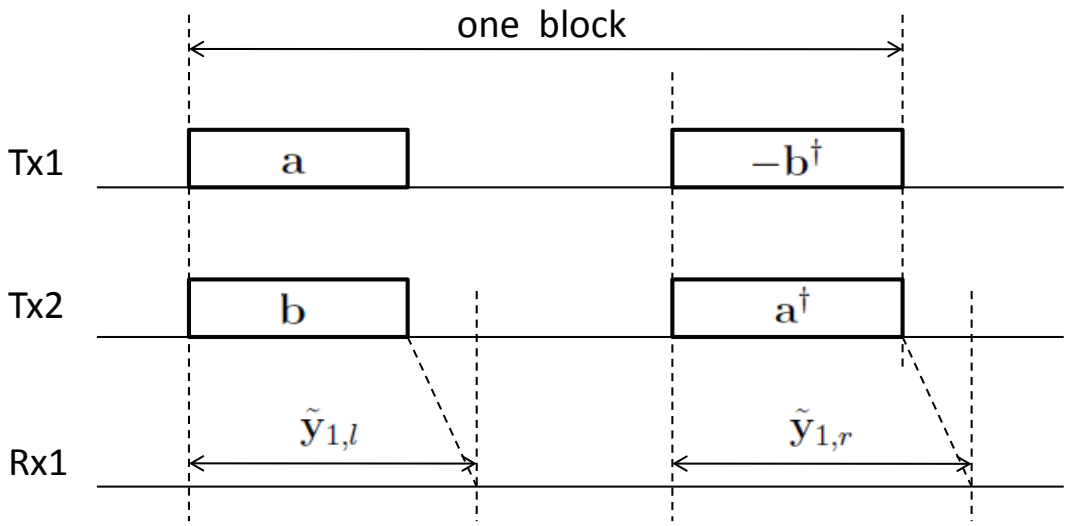


Figure 4-1. Alamouti diversity scheme of a system with 2 transmitters and 1 receiver.

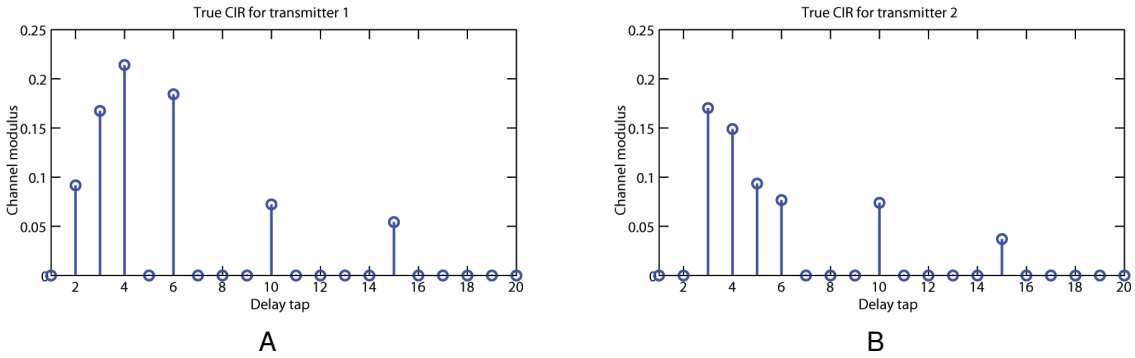


Figure 4-2. The modulus of the simulated CIRs between the two transmitters and the receiver.

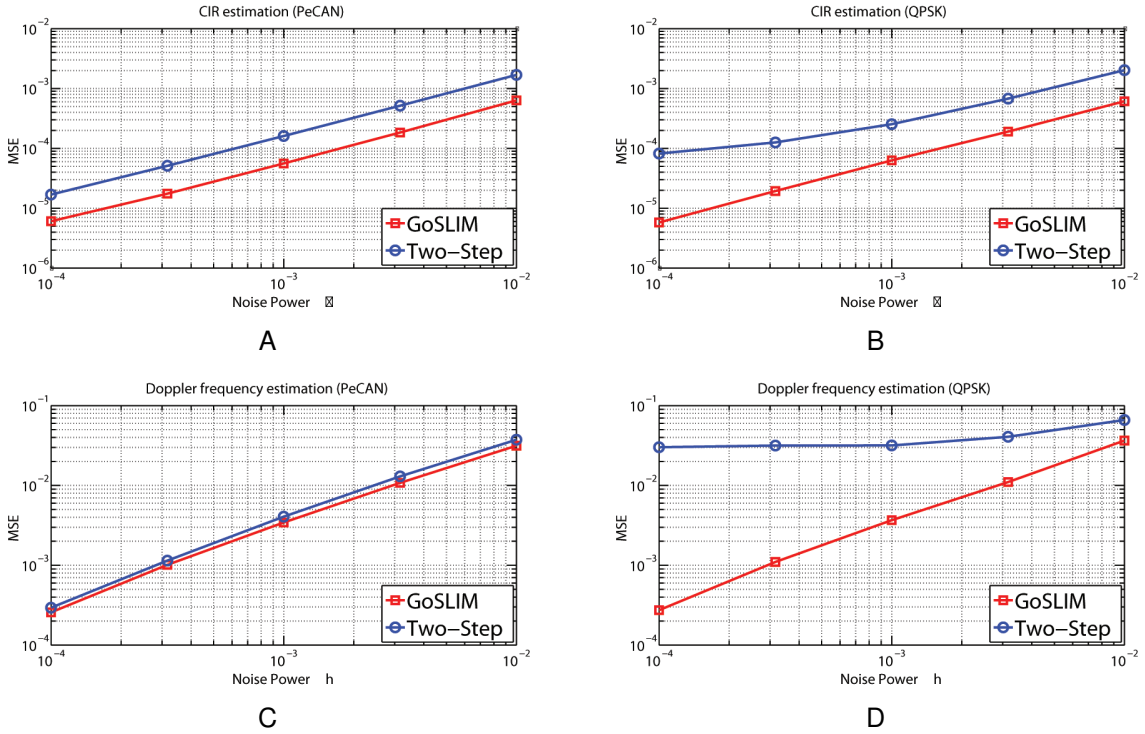


Figure 4-3. A) MSEs of CIR estimates for PeCAN sequences. B) MSEs of CIR estimates for QPSK sequences. C) MSEs of Doppler frequency estimates for PeCAN sequences. D) MSEs of Doppler frequency estimates for QPSK sequences. Both sequences have a length of 256 symbols. Each point is averaged over 100 Monte-Carlo trials.

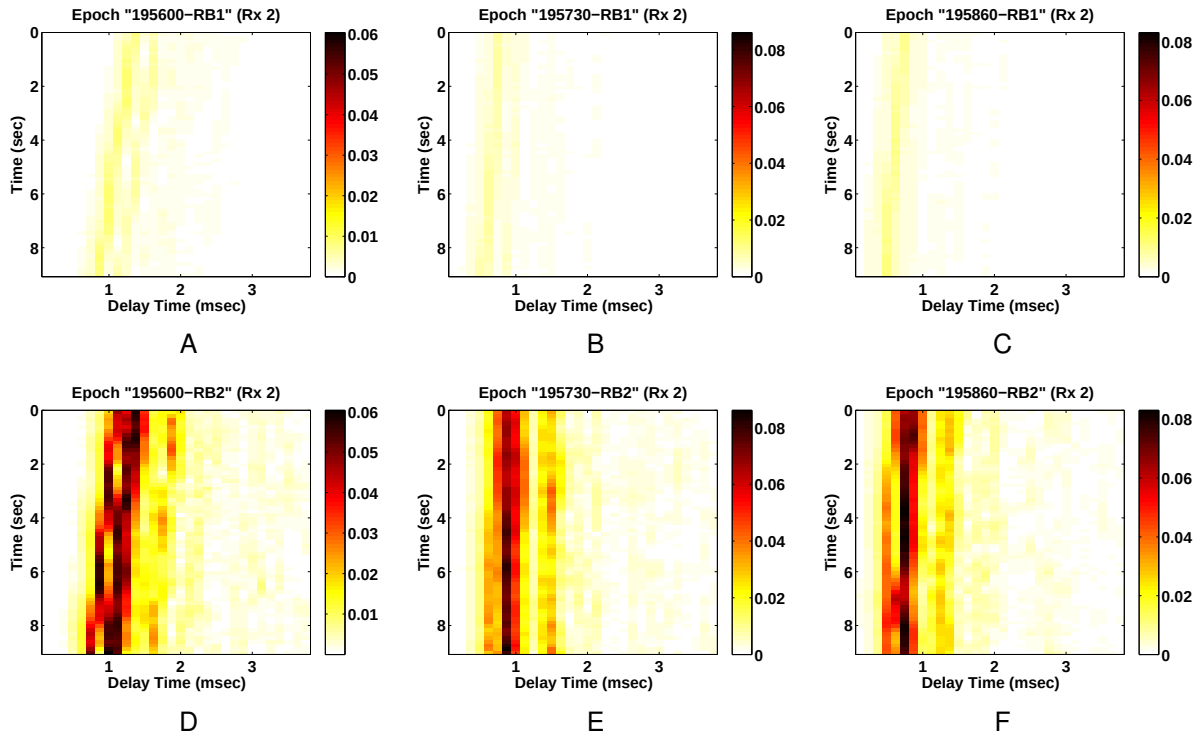


Figure 4-4. Evolutions of the CIRs between the active transmitter to the second receiver of the RB1 and RB2 arrays for all 6 epochs.

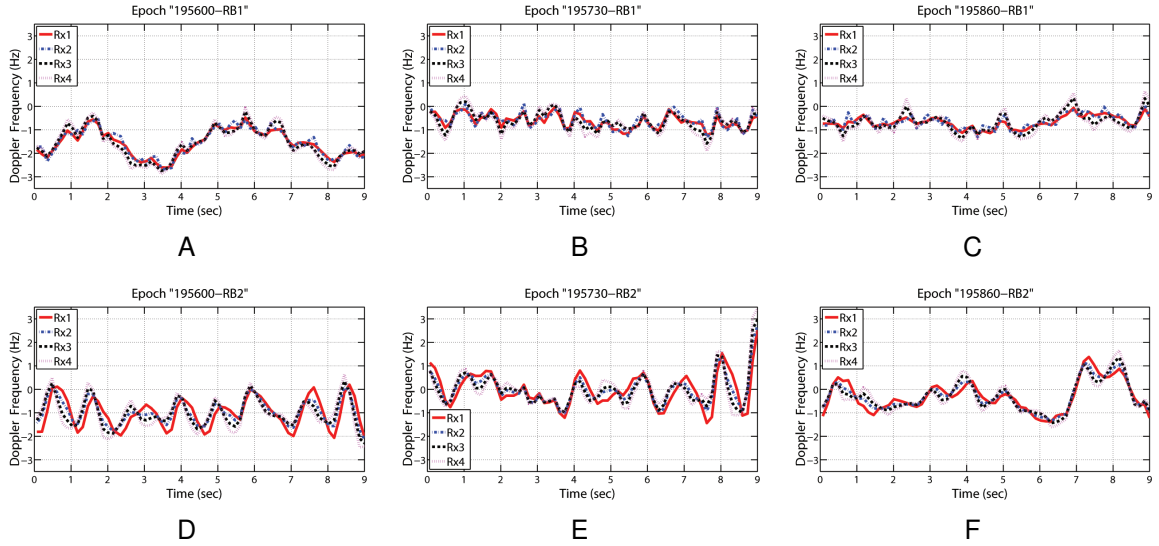


Figure 4-5. Evolutions of the estimated Doppler frequencies at each receiver for all 6 epochs.

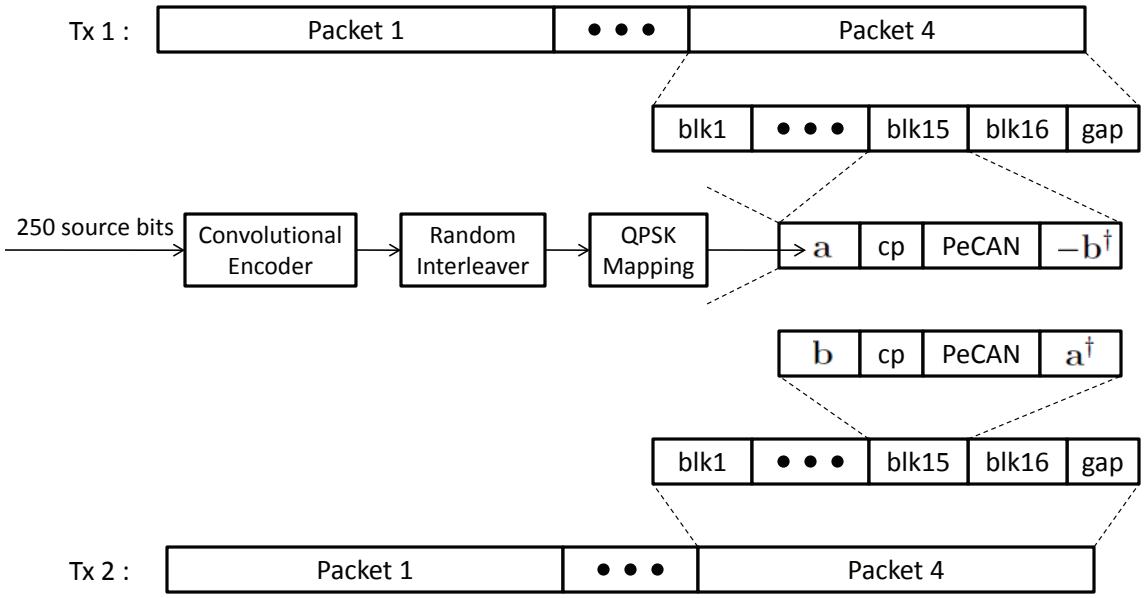


Figure 4-6. The structure of our transmitted symbols for the 2Tx−1Rx Alamouti scheme used in the WHOI09 experiment.

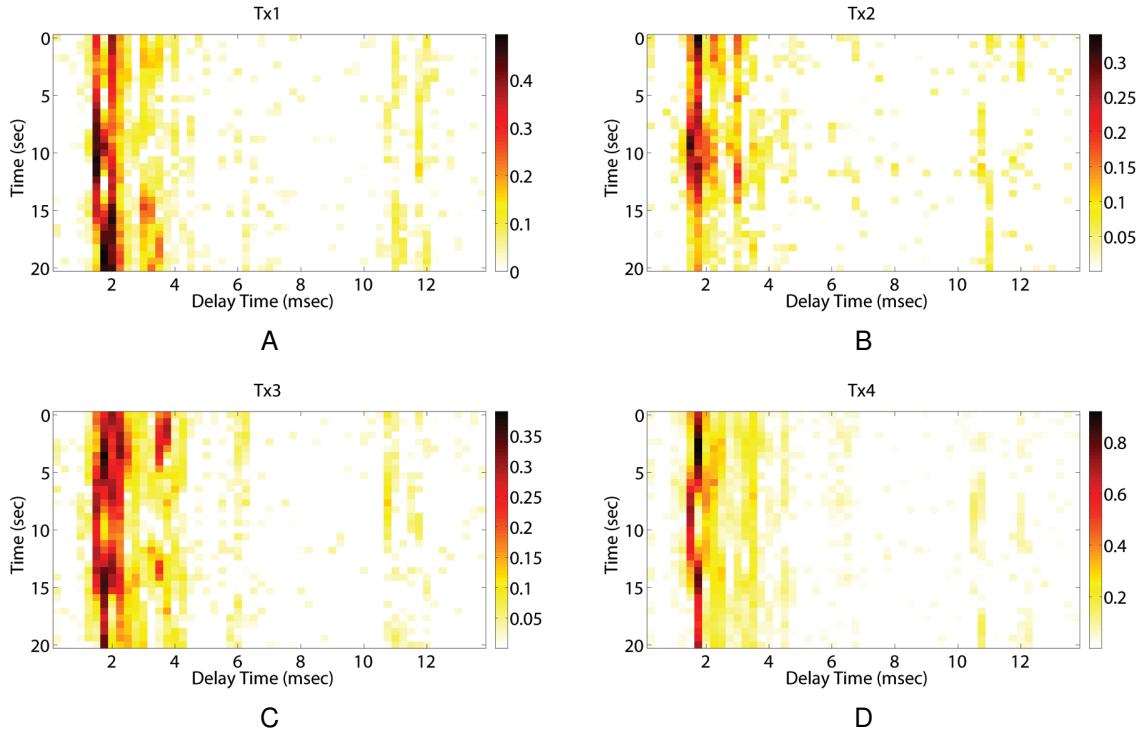
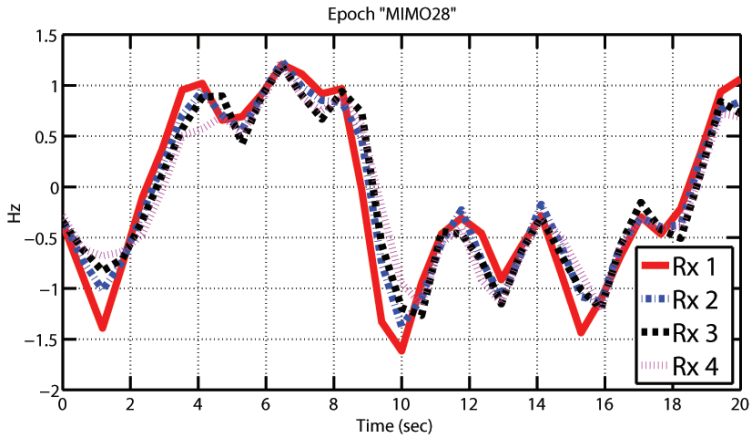
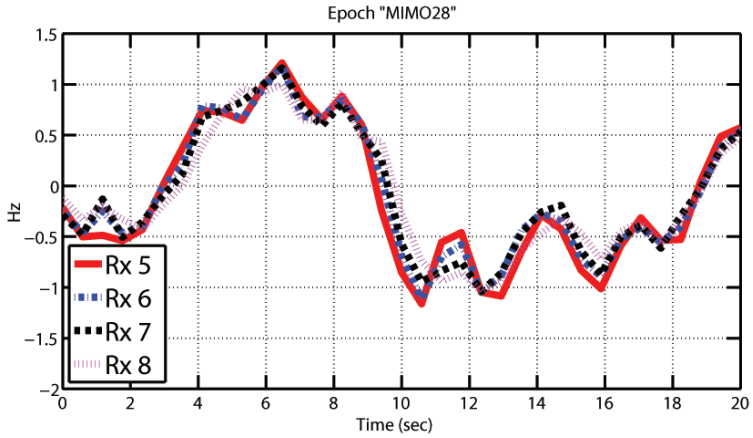


Figure 4-7. CIR evolutions between the four active transmitters and one hydrophone for Epoch “MIMO28”.



A



B

Figure 4-8. Evolutions of the estimated Doppler frequencies at each receiver for Epoch "MIMO28".

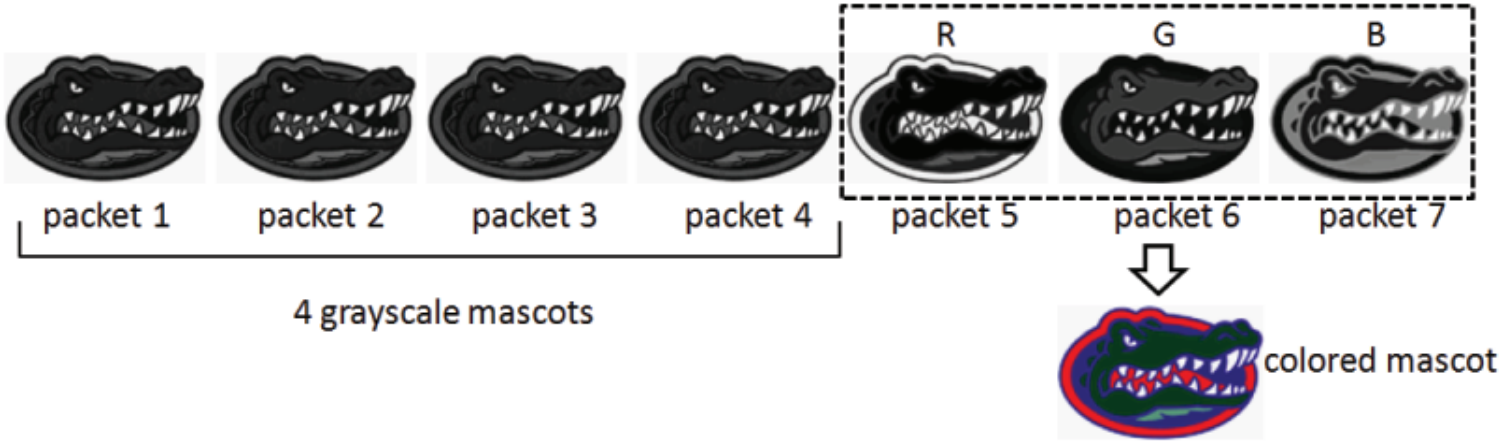


Figure 4-9. Each package transmitted in the ACOMM10 experiment contains 4 grayscale Gator mascot images and 1 colored image. The colored image is decomposed into RGB components.

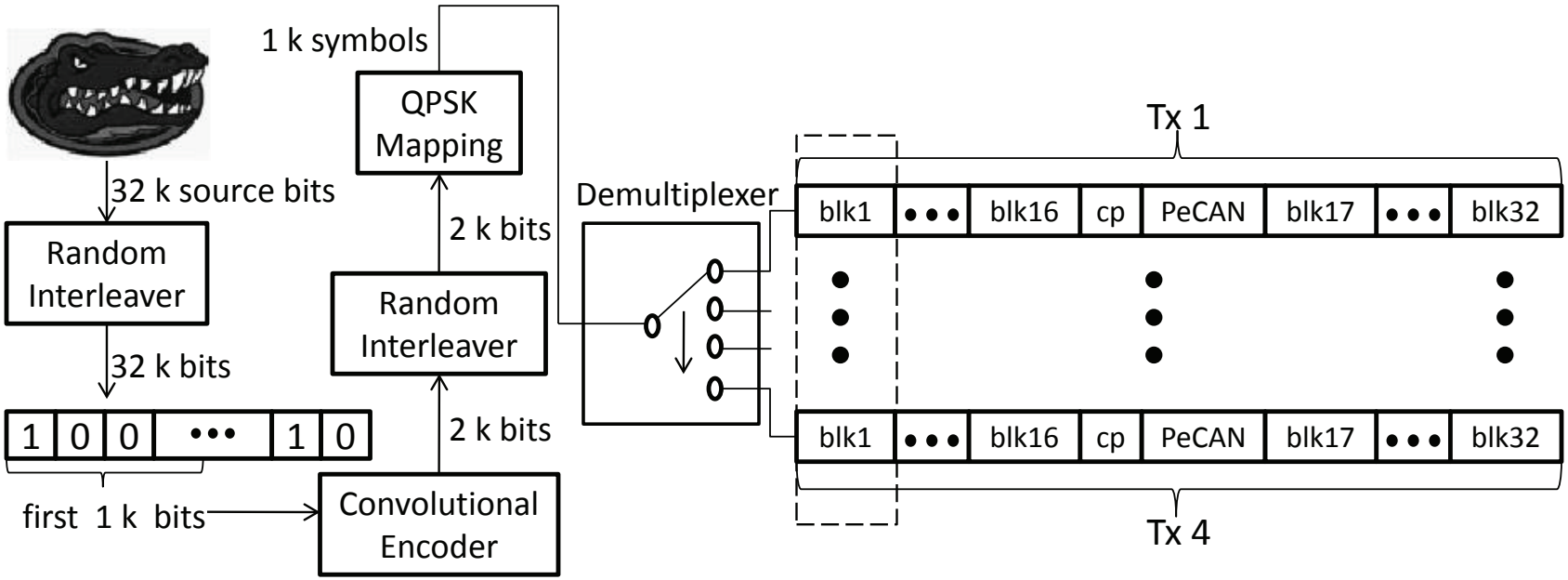


Figure 4-10. The structure of the transmitted symbols for the 4×8 MIMO BLAST scheme used in ACOMM10.

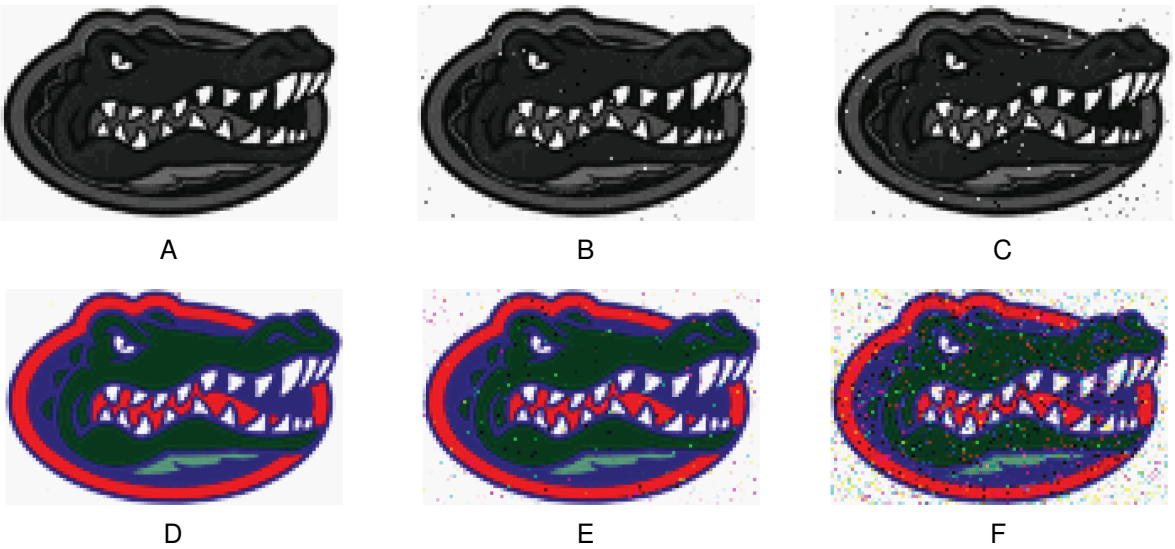


Figure 4-11. A) Grayscale mascot recovered from epoch “MIMO08”. B) Grayscale mascot recovered from epoch “MIMO03”. C) Grayscale mascot recovered from epoch “MIMO27”. D) Colored mascot recovered from epoch “MIMO08”. E) Colored mascot recovered from epoch “MIMO03”. F) Colored mascot recovered from epoch “MIMO27”.

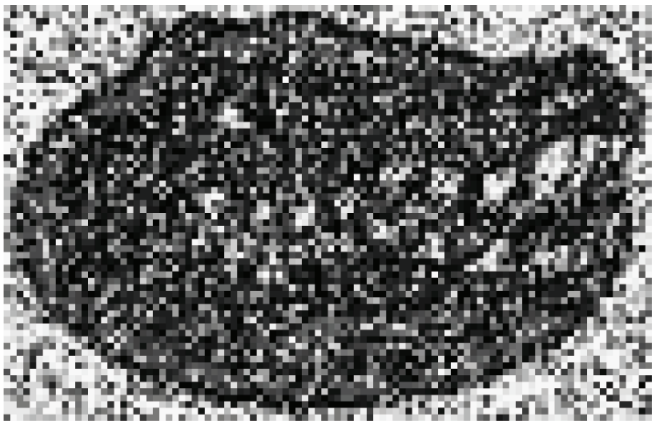


Figure 4-12. Grayscale mascot recovered from epoch “MIMO08” using SLIM.

CHAPTER 5
FUTURE WORK

In Chapters 2, 3, and 4, We have studied various critical problems that arise in practical MIMO UAC, such as the synthesis of the effective training sequences at the transmitter, the development of the channel estimation and symbol detection schemes at the receiver side, etc. The effectiveness of the proposed MIMO UAC techniques is verified using both numerical examples and several in-water experiments. The long-term goal of a UAC project is to implement sophisticated signal processing algorithms and coding techniques in a small-area, low-power device that can be equipped on a watercraft to achieve high speed and reliable UAC in real-time. In this chapter, our focus is shifted from research idea development to concrete system implementation by criticizing the existing MIMO UAC schemes from an application point of view. Moreover, we also provide a vision for the future of UAC by discussing the possibilities and challenges of employing multiuser techniques in the underwater environments.

5.1 MIMO UAC: An Application Point of View

5.1.1 Data Rate

The speed of a digital wireless service in terms of data rate is normally one of the most important specifications a customer would be concerned about. The highest data rate we had achieved during the participation in the MIMO UAC project was 62.5 Kbps, which was obtained by analyzing the 200 m measurements acquired on good channel conditions during the course of the SPACE08 in-water experiment. This data rate allows the system to transmit two grayscale Gator mascots (Figure 4-10) back-to-back in approximately one second. In sharp contrast, the current WLAN standard offers a maximum collective data rate at tens of Mbps (Table 1-1), fast enough to download a 5-minute video in one second. Although the past three decades have seen a tremendous increase in the data rate of digital UAC systems resulting from major technical breakthroughs including the employment of phase coherent

communication schemes and a MIMO system telemetry, to date, UAC speed is still not able to compete with that achieved via wireless radio communications mainly due to the unique challenges imposed by the underwater environments (Section 1.1). The data rate in MIMO UAC has to be further increased before the proposed MIMO UAC schemes can transition from the current research stage to practical military or commercial applications and make an impact on everyday life.

At a fixed symbol rate, data rate can be increased by trading off communication reliability in terms of BER. For example, we can let each symbol convey more information via sophisticated modulation schemes [55]. Besides the QPSK modulation scheme employed throughout the present dissertation in which one symbol carries two bits, we can consider mapping three bits to one symbol through 8-PSK modulation, or even mapping four bits through 16-QAM (quadrature amplitude modulation). 8-PSK and 16-QAM constellations are shown in Figures 5-1 and 5-2, respectively. At a fixed average transmit power, as a symbol carries more bits, the constellation points become more clustered (Figures 5-1 and 5-2), and the resulting performance of symbol detection is more vulnerable to ambient noise and/or interferences. Moreover, unlike the PSK modulation scheme, where only the phase of the transmitted symbol contains information, for QAM, the information bits are encoded in both the amplitude and phase of the symbol; see Figure 5-2. Consequently, QAM modulated signals do not have constant envelop, which is an undesired feature from an amplifier efficiency point of view. Another preferable means to increase the data rate is to adopt an encoder with a higher rate via puncturing, or to completely skip the channel coding stage (i.e., directly transmitting uncoded information) at the cost of reduced error correction ability, or none at all [55]. In addition, we can also equip more transmitters to send signals simultaneously over the acoustic channel. The resulting detection performance for a particular transmitter is expect to degrade owing to the increased level

of interferences. On top of that, both the system cost and the receiver complexity will increase accordingly.

5.1.2 Real-Time Implementation

Different applications have different delay constraints. Services like email or paging in general can tolerate a considerable amount of delay, whereas for other applications, such as live broadcast or voice services, real-time constraint needs to be enforced. (In voice systems, a delay larger than 0.1 s could be noticed by the end user.) As a consequence, whether to address the real-time requirement or not depends on the specific application at hand.

For typical UAC applications, such as tactical communications between submarines in the battlefield or between UUVs engaged in a complex and dangerous task (oil spill control, for instance), each watercraft is required to interpret an order and respond promptly before the best chance dashes away. Therefore, real-time reception indeed is a critical factor on which the overall UAC system performance depends.

Based on the current state of hardware, it is feasible to develop an embedded system to achieve reliable UAC reception in real-time. For instance, DFE, coupled with LMS or RLS algorithm for updating the filter coefficients involved, is a preferable receiver structure to realize real-time. The Alamouti diversity scheme presented in Section 4.2.3 is also amenable to real-time implementation on a hardware platform. Both systems, however, do not fall into the MIMO UAC category and the corresponding data rate is in general much lower than could be achieved by MIMO UAC systems.

However, many technical challenges remain in implementing the proposed MIMO UAC schemes on a hardware platform to realize high-speed (with respect to the data rate achieved by DFE and Alamouti diversity scheme), reliable UAC in real-time. The bottleneck comes from both the limitation of the hardware and the mechanism of the algorithm. It is well known that the frequency of a central processing unit, the memory capacity, and the width of the data bus are among the key factors

that determine the overall speed of the resulting embedded system. Advances in hardware techniques make the digital hardwares more affordable, have faster speed and enhanced performance, and will continue to herald novel and improved signal processing approaches in MIMO UAC applications.

In an algorithm development perspective, it is preferable to simplify an algorithm as much as possible before transplanting it to a hardware platform. The efficient calculation of the LMMSE filter coefficients elaborated in Section 3.3.2 is a good example in this respect. Specifically, the CG method is employed to solve a linear system in an iterative manner, which avoids inverting a matrix with large dimensions. Moreover, the computations involved in each CG iteration are significantly expedited by making use of FFT and IFFT operations. This way, the original computationally expensive problem is decomposed into multiple basic operations that can be easily handled by normal hardwares. In addition, a hardware processing unit, such as digital signal processor (DSP) or field-programmable gate array (FPGA), is normally designed, or can be configured, to have multiple sets of the same functional unit. These parallel units can operate simultaneously without interfering with each other. Therefore, a DSP- or FPGA-friendly algorithm is one that exploits parallel computations. Although high computations of IAA makes it unattractive for hardware implementation, its parallel updating procedure elaborated in Section 2.2.3 can still be appreciated.

5.2 Multiuser UAC Systems

The focus of this dissertation is on MIMO UAC from one user to the other. A defining characteristic of such a communication scheme is that there is only one active user involved and the transmitted signal occupies all the resources the channel provides, including time and bandwidth. The explosive growth of the cellular systems featuring multiuser techniques (where multiple users share the available resources) drives us to briefly study the possibilities and challenges of implementing a similar multiuser system in the underwater environments.

Multiuser systems involve two types of channels: a downlink channel over which one transmitter (a based station) sends information to many receivers (end users), and an uplink channel over which many transmitters (end users) send information to one receiver (a based station) [21]. The most common methods to divide the available resources are along the frequency, time, and code axes. The resulting multiple access techniques are respectively referred to as frequency-division multiple access (FDMA), time-division multiple access (TDMA), and code-division multiple access (CDMA), which we will study in sequel.

5.2.1 Frequency-Division Multiple Access

In FDMA systems, as shown in Figure 5-3, the available bandwidth is divided into multiple channels, and each user is assigned a different channel. To ensure the absence of cross-channel interferences, channels do not overlap with each other. The transmission in FDMA systems is continuous over time, and a frequency modulation module is needed to tune the signal to a specified carrier frequency associated with the channel.

If the FDMA scheme takes place in realistic underwater acoustic environment, then additional challenges are present. As we previously remarked in Section 1.1, the available bandwidth offered by the acoustic channels is rather scarce: 40 KHz at most compared to 20 MHz for WLAN. This essentially limits the number of users an underwater FDMA system can accommodate. Moreover, the channel-induced Doppler effects will shift the signal bandwidth during prorogation [55]. To see this, suppose a watercraft is approaching a stationary base station at a velocity of v m/s, and it is sending a sinusoid signal. Due to the Doppler effects, the transmitted sinusoid at frequency f will be converted to a sinusoid of frequency of $\left(1+\frac{v}{c}\right) f$ at the receiver side, in which c represents the underwater sound speed. The frequency increment $\frac{vf}{c}$ is commonly referred to as Doppler shift. In a similar manner, if the watercraft is moving away from the base station, the frequency at the receiver becomes $\left(1-\frac{v}{c}\right) f$ with a

Doppler shift $-\frac{vf}{c}$. To mitigate such frequency shift and the imperfect pulse shaping filter, a guard band should be allocated between adjacent channels.

5.2.2 Time-Division Multiple Access

TDMA, as the name suggests, segments the continuous time axis into non-overlapping time slices of equal length; see Figure 5-4. The time slices are assigned to each user in equal portions and in circular order (this scheduling scheme is also referred to as a round-robin scheduling in computer network community). In TDMA systems all users share the available frequency bandwidth, and the transmission is not continuous over time. The round-robin scheduling scheme guarantees that at any time there is at most one user transmitting a signal, which eliminates the cross-user interferences.

Two critical problems need to be addressed before a TDMA system can be successfully implemented in the underwater environments. The first problem is how to achieve synchronization among all the users. As just mentioned, the suppression of the cross-user interferences in TDMA systems is completely realized via timing: each user needs to know the exact time to switch the transmission on and off. For radio communications, synchronization can be simply achieved with the help of a global positioning system (GPS) signal. However, a GPS signal experiences trouble penetrating through the water medium, which leads us to developing other feasible means to achieve synchronization in the underwater environments. The second problem is the presence of severe ISI due to the long CIR (Section 1.1). In ideal flat-fading channels without ISI, right after the transmission of the current user is switched off, the next user in line can start transmitting immediately: no time is wasted during the transition. However, in realistic frequency selective channels with the presence of the ISI, to ensure the absence of cross-user interferences, the next user must postpone its transmission until the signal sent by the current user completely dies out. In other words, guard intervals are necessary between adjacent time slices, and the length of the guard intervals should be longer than the channel delay spread (typically in tens of ms). Since

no signal is transmitted during guard intervals, a considerable amount of time will be wasted, especially when the length of the guard interval becomes a significant fraction of the length of each time slice.

5.2.3 Code-Division Multiple Access

In CDMA systems different users employ different spreading sequences to modulate the information signal before transmission, and the modulated signals share the same time and bandwidth resources available; see Figure 5-5. The level of the cross-user interference in CDMA systems is mainly determined by the correlation properties of the spreading sequences and the characteristics of the underlying channel [21]. For example, in an ideal flat-fading downlink channel with perfect synchronization, the Hadamard sequences, which are orthogonal to each other, could be the optimal spreading sequences in terms of eliminating all possible interferences [60]. However, in a frequency selective downlink channel, Hadamard sequences are no longer the optimal choice. Instead, spreading waveforms with good auto- and cross-correlations over certain lags are preferred. These types of sequences are usually referred to as zero-correlation zone sequences and the relevant literature is extensive [17, 78, 80].

In the uplink scenario, on the other hand, it is natural to allow each user to transmit signals to the base station at any convenient time, and therefore, the requirement of synchronization among users cannot, and should not, be enforced. The relaxation of the synchronization requirement amounts to synthesizing spreading sequences with good auto- and cross-correlation properties over the entire lags, such as the Gold sequences and Kasami sequences [84].

When CDMA is implemented in the realistic underwater environments, the time-varying nature of the acoustic channel prefers the employment of short spreading sequences since otherwise the block fading assumption can be easily violated [45]. One top of that, the presence of Doppler effects also induces temporal scaling (stretching or compression) to the transmitted signals. Doppler induced scaling effects can

easily destroy the orthogonality properties of the spreading sequences. To address this problem, it is desirable to transmit Doppler-sensitive probing sequences with good ambiguity functions (ideally, thumbtack-like auto-ambiguity functions and zero cross-ambiguity functions). Due to the extreme difficulty of the problem, it is still an open question as to how to synthesize such desired spreading sequences effectively and efficiently [35].

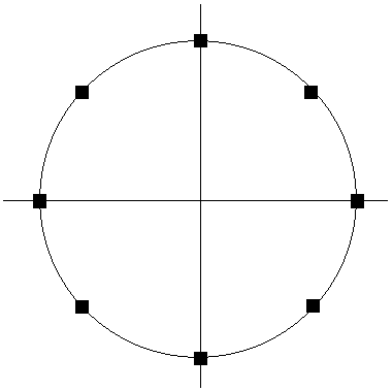


Figure 5-1. 8-PSK constellation.

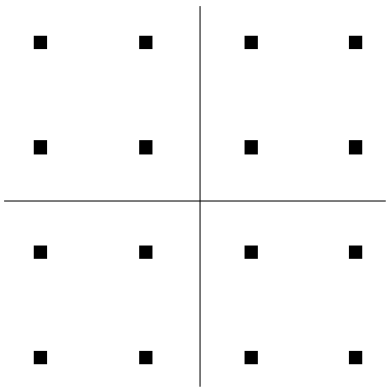


Figure 5-2. 16-QAM constellation.

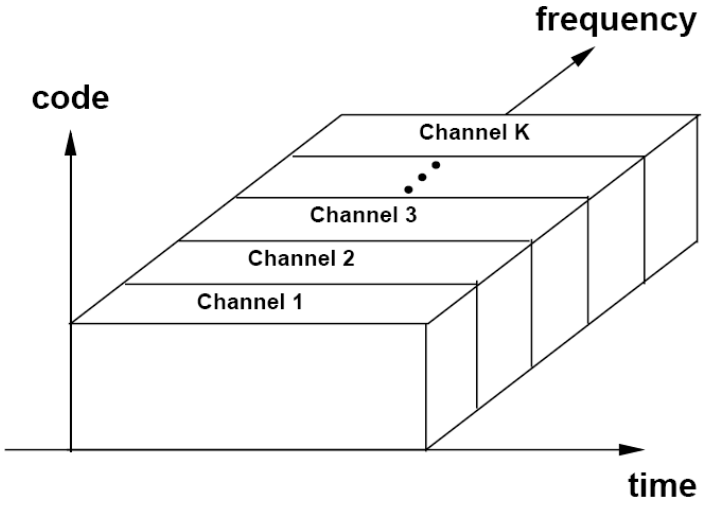


Figure 5-3. Frequency-division multiple access (FMDA). Copyright image courtesy of [21].

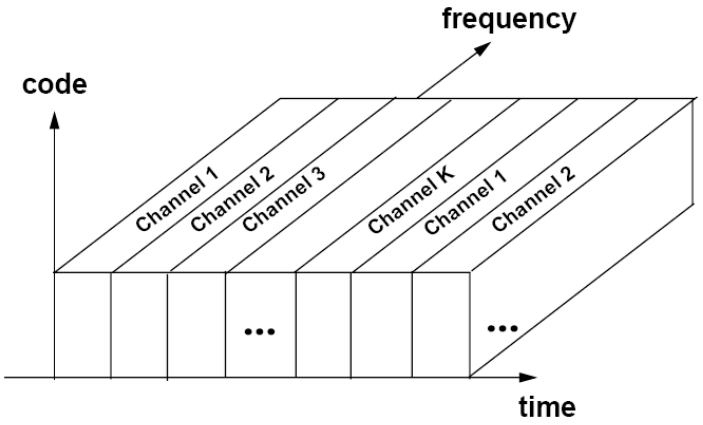


Figure 5-4. Time-division multiple access (TDMA). Copyright image courtesy of [21].

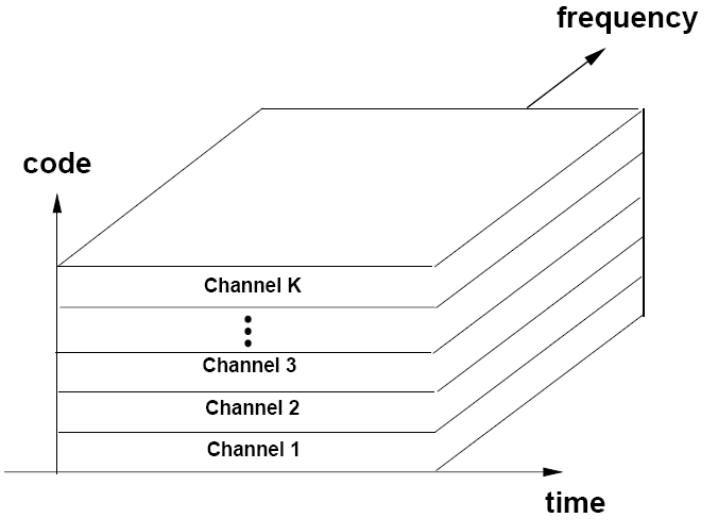


Figure 5-5. Code-division multiple access (CDMA). Copyright image courtesy of [21].

REFERENCES

[1] Al-Dhahir, N. "Single-Carrier Frequency-Domain Equalization for Space-Time Block-Coded Transmissions Over Frequency-Selective Fading Channels." *IEEE Communications Letters* 5 (2001).7: 304–306.

[2] Alamouti, S. M. "A simple transmit diversity techniques for wireless communications." *IEEE Journal on Selected Areas in Communications* 16 (1998).8: 1451–1458.

[3] Appleton, E. "Automatic synchronization of triode oscillators." *Proc. Cambridge Phil. Soc.* 21 (1922–1923): 231–248.

[4] Baggeroer, A. B., Koelsch, D., von der Heydt, K., and Catipovic, J. "DATS - A Digital Acoustic Telemetry System for Underwater Communications." *Proceedings of Oceans,* Boston, MA (1981): 55–61.

[5] Carbonelli, C., Vedantam, S., and Mitra, U. "Sparse Channel Estimation with Zero Tap Detection." *IEEE International Conference on Communications* Paris, France 6 (2004): 3173–3177.

[6] Catipovic, J. A. "Performance limitations in underwater acoustic telemetry." *IEEE Journal of Oceanic Engineering* 15 (1990).3: 205–216.

[7] Chen, R.-R. and Peng, R. "Noncoherent detection based on Markov Chain Monte Carlo methods for block fading channels." *in Proc. IEEE Global Telecommunications Conference,* St. Louis, MO 5 (2005): 2517–2521.

[8] Cotter, S. F., Adler, R., Rao, R. D., and Kreutz-Delgado, K. "Forward sequential algorithms for best basis selection." *IEE Proceedings of Vision, Image and Signal Processing* 146 (1999).5: 235–244.

[9] Cotter, S. F. and Rao, B. D. "The adaptive matching pursuit algorithm for estimation and equalization of sparse time-varying channels." *25th Asilomar Conference on Signals, Systems and Computers,* Pacific Grove, CA 2 (2000): 1772–1776.

[10] ———. "Sparse channel estimation via matching pursuit with application to equalization." *IEEE Transactions on Communications* 50 (2002).3: 374–377.

[11] Cupo, R. L., Golden, G. D., Martin, C. C., Sherman, K. L., Sollenberger, N. R., Winters, J. H., and Wolniansky, P. W. "A Four-Element Adaptive Antenna Array for IS-136 PCS Base Stations." *47th IEEE Vehicular Technology Conference* 3 (1997): 1577–1581.

[12] Douillard, C., Jezequel, C. M., Berrou, C., Picart, A., Didier, P., and Glavieux, A. "Iterative correction of intersymbol interference: Turbo-equalization." *European Transactions on Telecommunications* 6 (1995): 507–511.

[13] Dowling, D. R. "Acoustic pulse compression using passive phase-conjugate processing." *Journal of the Acoustical Society of America* 95 (1994): 1450–1458.

[14] Eggen, T. H., Baggeroer, A. B., and Preisig, J. C. "Communication Over Doppler Spread Channels – Part I: Channel and Receiver Presentation." *IEEE Journal of Oceanic Engineering* 25 (2000).1: 62–71.

[15] Eggen, T. H., Preisig, J. C., and Baggeroer, A. B. "Communication Over Doppler Spread Channels – Part II: Receiver characterization and practical results." *IEEE Journal of Oceanic Engineering* 26 (2001).4: 612–621.

[16] Fan, P. and Mow, W. "On Optimal Training Sequence Design for Multiple-Antenna Systems Over Dispersive Fading Channels and Its Extensions (Correspondence)." *IEEE Transactions Vehicular Technology* 53 (2004).5: 1623–1626.

[17] Fan, P., Suehiro, N., Kuroyanagi, N., and Deng, X. "Class of binary sequences with zero correlation zone." *Electronics Letters* 35 (1999): 777–779.

[18] Flynn, J. A., Fox, W. L. J., Ritcey, J. A., and Rouseff, D. "Performance of reduced-complexity multi-channel equalizers for underwater acoustic communications." *in Proceedings of the 36th Asilomar Conference* 1 (2002): 453–460.

[19] Freitag, L., Johnson, M., and Stojanovic, M. "Efficient Equalizer Update Algorithms for Acoustic Communication Channels of Varying Complexity." *IEEE/MTS Oceans Conference* 1 (1997): 580–585.

[20] Garrood, D. J. "Application of the MFSK acoustical communication system." *Proceedings of Oceans,* Boston, MA (1981): 67–71.

[21] Goldsmith, A. *Wireless Communications*. 32 Avenue of the Americans, New York, NY: Cambridge University Press, 2005.

[22] Golub, G. H. and Loan, C. F. Van. *Matrix Computations*. Baltimore, MD: Johns Hopkins University Press, 1989.

[23] Gray, S. D., Preisig, J. C., and Brady, D. "Multiuser detection in a horizontal underwater acoustic channel using array observations." *IEEE Transactions on Signal Processing* 45 (1997).3: 148–160.

[24] Groh, I., Dietl, G., and Utschick, W. "Preconditioned and rank-flexible block conjugate gradient implementations of MIMO Wiener decision feedback equalizers." *in Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP,* Toulouse, France 4 (2006): 441–444.

[25] Guerci, J. R., Goldstein, J. S., and Reed, I. S. "Optimal and adaptive reduced-rank STAP." *IEEE Transactions on Aerospace and Electronic Systems. (Special Section on Space-Time Adaptive Processing)* 36 (2000).2: 647–663.

[26] He, H., Vu, D., Stoica, P., and Li, J. "Construction of Unimodular Sequence Sets for Periodic Correlations." *43rd Annual Asilomar Conference on Signals, Systems and Computers,* Pacific Grove, CA (2009): 136–140.

[27] Högbom, J. A. "Aperture synthesis with a non-regular distribution of interferometer baselines." *Astronomy and Astrophysics Supplements* 15 (1974.): 417–426.

[28] Horn, R. A. and Johnson, C. R. *Matrix Analysis*. Cambridge, U.K.: Cambridge University Press, 1985.

[29] Hsieh, G. and Hung, J. C. "Phase-Locked Loop Techniques-A Survey." *IEEE Transactions on Industrial Electronics* 43 (1996): 609–615.

[30] Johnson, M., Freitag, L., and Stojanovic, M. "Improved Doppler Tracking and Correction for Underwater Acoustic Communication." *IEEE International Conference on Acoustics, Speech, and Signal Processing,* Munich, Germany 1 (1997): 575–578.

[31] Kilfoyle, D. B. and Baggeroer, A. B. "The State of the Art in Underwater Acoustic Telemetry." *IEEE Journal of Oceanic Engineering* 25 (2000): 4–27.

[32] Kilfoyle, D. B., Preisig, J. C., and Baggeroer, A. B. "Spatial modulation experiments in the underwater acoustic channel." *IEEE Journal of Oceanic Engineering* 30 (2005).2: 406–415.

[33] Kocic, M., Brady, D., and Stojanovic, M. "Sparse equalization for real-time digital underwater acoustic communications." *IEEE/MTS Oceans Conference* 3 (1995): 1417–1422.

[34] Koike, T. "Optimum-weighted RLS channel estimation for time-varying fast fading MIMO channels." *IEEE International Conference on Communications,* Glasgow, Scotland (2007): 24–28.

[35] Levanon, Nadav and Mozeson, Eli. *Radar Signals*. NY: Wiley, 2004.

[36] Li, B., Zhou, S., Milica, S., Freitag, L., Huang, J., and Willett, P. "MIMO-OFDM over and underwater acoustic channel." *in Proceedings of the Oceans,* Vancouver, BC, Canada (2007): 1–6.

[37] Li, J. and Stoica, P. "An adaptive filtering approach to spectral estimation and SAR imaging." *IEEE Transactions on Signal Processing* 44 (1996).6: 1469–1484.

[38] ———. "Efficient mixed-spectrum estimation with applications to target feature extraction." *IEEE Transactions on Signal Processing* 44 (1996): 281–295.

[39] Li, J., Stoica, P., and Zheng, D. "Angle and waveform estimation via RELAX." *IEEE Transactions on Aerospace and Electronic Systems* 33 (1997): 1077–1087.

[40] Li, J., Zheng, X., and Stoica, P. "MIMO SAR Imaging: Signal Synthesis and Receiver Design." *The 2nd International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (invited),* St. Thomas, U.S. Virgin Islands (2007).

[41] ———. "Signal Synthesis and Receiver Design for MIMO Radar Imaging." *IEEE Transactions on Signal Processing* 56 (2008).8: 3959–3968.

[42] Li, W. "Estimation and Tracking of Rapidly Time-Varying Broadband Acoustic Communication Channels." *PhD thesis* (Massachusetts Institute of Technology, Cambridge, MA, 2005).

[43] Li, W. and Preisig, J. C. "Estimation of Rapidly Time-Varying Sparse Channels." *IEEE Journal of Oceanic Engineering* 32 (2007).4: 927–939.

[44] Lindskog, E. and Paulraj, A. "A transmit diversity scheme for channels with intersymbol interference." *Proceedings of ICC 2000* 1 (New Orleans, LA, June 2000): 307–311.

[45] Ling, J., He, H., Stoica, P., Li, J., and Roberts, W. "Covert Underwater Acoustic Communications." *Journal of the Acoustical Society of America* 128 (2010): 2898–2909.

[46] Ling, J., Tan, X., Yardibi, T., Li, J., Nordenvaad, M. L., and He, H. "Enhanced channel estimation and efficient symbol detection in MIMO underwater acoustic communications." *43rd Annual Asilomar Conference on Signals, Systems, and Computers,* Pacific Grove, CA (2009).

[47] Ling, J., Yardibi, T., Su, X., He, H., and Li, J. "Enhanced channel estimation and symbol detection for high speed MIMO underwater acoustic communications." *IEEE 13th DSP Workshop & 5th SPE Workshop,* Marco Island, FL (2009): 600–604.

[48] ———. "Enhanced channel estimation and symbol detection for high speed multi-input multi-ouput underwater acoustic communications." *Journal of Acoustic Society of America* 125 (2009).5: 3067–3078.

[49] Madhow, U. and Honig, M. L. "MMSE interference suppression for direct-sequence spread-spectrum CDMA." *IEEE Transactions on Communications* 42 (1994).12: 3178–3188.

[50] Mallat, S. and Zhang, Z. "Matching Pursuits with time-frequency dictionaries." *IEEE Transactions on Signal Processing* 41 (1993).12: 3397–3415.

[51] Miller, S. L. and O'Dea, R. J. "Peak Power and Bandwidth Efficient Linear Modulation." *IEEE Transactions on Communications* 46 (1998).12: 1639–1648.

[52] Natarajan, B. K. "Sparse approximation solutions to linear systems." *SIAM Journal of Computing* 24 (1995): 227–234.

[53] Nordenvaad, M. and Oberg, T. “Iterative reception for acoustic underwater MIMO communications.” *in Proceedings of the Oceans,* Boston, MA 21 (2006).3: 1–6.

[54] Poor, H. V. and Verdú, S. “Probability of error in MMSE multiuser detection.” *IEEE Transactions on Information Theory* 43 (1997).3: 858–871.

[55] Preisig, J. “Acoustic propagation considerations for underwater acoustic communications network development.” *SIGMOBILE Mobile Computing Communications Review* 11 (2007).4: 2–10.

[56] Preisig, J. C. “Performance analysis of adaptive equalization for coherent acoustic communications in the time-varying ocean environment.” *Journal of the Acoustical Society of America* 118 (2005).1: 263–278.

[57] Proakis, J. G. *Digital Communications*. McGraw-Hill Inc., third edition, 1995.

[58] Robertson, P., Villebrun, E., and Hoeher, P. “A comparison of optimal and sub-optimal MAP decoding algorithms operating in the log domain.” *IEEE International Conference on Communications,* Seattle, WA (1995): 1009–1013.

[59] Roy, S., Duman, T. M., McDonald, V., and Proakis, J. G. “High-rate communication for underwater acoustic channels using multiple transmitters and space-time coding: Receiver structures and experimental results.” *IEEE Journal of Oceanic Engineering* 32 (2007): 663–688.

[60] Ryser, H. J. *Combinatorial Mathematics*. John Wiley and Sons, 1963.

[61] Sayed, A. H. *Fundamentals of Adaptive Filtering*. New York, NY: John Wiley and Sons, Inc., 2003.

[62] Scharf, L. L., Chong, E. K. P, Zoltowski, M. D., Goldsein, J. S., and Reed, I. S. “Subspace Expansion and the Equivalence of Conjugate Direction and Multistage Wiener Filters.” *IEEE Transactions on Signal Processing* 56 (2008): 5013–5019.

[63] Schwarz, G. “Estimating the Dimension of a Model.” *The Annals of Statistics* 6 (1978): 461–464.

[64] Schwarz, U. J. “Mathematical-statistical Description of the Iterative Beam Removing Technique (Method CLEAN).” *Astronomy and Astrophysics* 65 (1974): 345–356.

[65] Song, A., Badiey, M., and McDonald, V. K. “Multichannel combining and equalization for underwater acoustic MIMO channels.” *in Proceedings of MTS/IEEE Oceans conference,* Quebec City, Canada (2008): 1–6.

[66] Song, B. and Ritcey, J. “Spatial diversity equalization for MIMO ocean acoustic communication channels.” *IEEE Journal of Oceanic Engineering* 21 (1996).3: 505–512.

[67] Stoica, P., Jakobsson, A., and Li, J. “Capon, APES and Matched-Filterbank Spectral Estimation.” *Signal Processing* 66 (1998).1: 45–59.

[68] Stoica, P., Li, H., and Li, J. “A new derivation of the APES filter.” *IEEE Signal Processing Letter* 6 (1999).8: 205–206.

[69] Stoica, P. and Moses, R. L. *Spectral Analysis of Signals*. Upper Saddle River, NJ: Prentice-Hall, 2005.

[70] Stoica, P. and Selén, Y. “Model-order selection: a review of information criterion rules.” *IEEE Signal Processing Magazine* 21 (2004).4: 36–47.

[71] Stoica, Petre, He, Hao, and Li, Jian. “On Designing Sequences with Impulse-like Periodic Correlation.” *IEEE Signal Processing Letters* 16 (2009).8: 703–706.

[72] Stojanovic, M. “Coherent digital communications for rapidly fading channels with applications to underwater acoustics.” Ph.D. dissertation, Northeastern University, Boston, MA (1993).

[73] ———. “Recent advances in high-speed underwater acoustic communications.” *IEEE Journal of Oceanic Engineering* 21 (1996).2: 125–136.

[74] Stojanovic, M., Catipovic, J., and Proakis, J. “Phase-coherent Digital Communications for Underwater Acoustic Channels.” *IEEE Journal of Oceanic Engineering* 19 (1994).1: 100–111.

[75] Stojanovic, M., Freitag, L., and Johnson, M. “Channel-Estimation-Based Adaptive Equalization of Underwater Acoustic Signals.” *IEEE/MTS Oceans Conference* 2 (1999): 985–990.

[76] Stojanovic, M. and Preisig, J. “Underwater Acoustic Communication Channels: Propagation Models and Statistical Characterization.” *IEEE Communications Magazine* (2009): 84–89.

[77] Strang, G. “A proposal for Toeplitz matrix calculations.” *Stud. Appl. Math.* 74 (1986): 171–176.

[78] Suehiro, N. “A signal design without co-channel interference for approximately synchronized CDMA systems.” *IEEE Journal on Selected Areas in Communications* 12 (1994).5: 837–841.

[79] Tipping, Michael E. “Sparse Bayesian Learning and the Relevance Vector Machine.” *Journal of Machine Learning Research* 1 (2001): 211–244.

[80] Torii, H., Nakamura, M., and Suehiro, N. “A new class of zero-correlation zone sequences.” *IEEE Transactions on Information Theory* 50 (2004).3: 559–565.

[81] Tropp, J., Dhillon, I., Heath, R., and Strohmer, T. "Designing structured tight frames via an alternating projection method." *IEEE Transactions on Information Theory* 51 (2005).1: 188–209.

[82] Tropp, J. A. "Just relax: Convex programming methods for identifying sparse signals." *IEEE Transactions on Information Theory* 51 (2006).3: 1030–1051.

[83] Tsai, P., Kang, H., and Chiueh, T. "Joint weighted least-squares estimation of carrier-frequency offset and timing offset for OFDM systems over multipath fading channels." *IEEE Transactions on Vehicular Technology* 54 (2005).1: 211–223.

[84] Tse, D. and Viswanath, P. *Fundamentals of Wireless Communication*. New York, NY: Cambridge University Press, 2005.

[85] Tseng, P. and Yun, S. "A coordinate gradient descent method for nonsmooth separable minimization." *Mathematical Programming* 117 (2007): 387–423.

[86] Vajapeyam, M., Vedantam, S., Mitra, U., Preisig, J., and Stojanovic, M. "Distributed Space-Time Cooperative Schemes for Underwater Acoustic Communications." *IEEE Journal of Oceanographic Engineering* 33 (2008).4: 489–501.

[87] van Nee, R., Awater, G., Morikura, M., Takanashi, H., Webster, M., and Halford, K. "New high-rate wireless LAN standards." *IEEE Communications Magazine* 37 (December 1999): 82–88.

[88] Van Trees, H. L. *Optimum Array Processing: Part IV of Detection, Estimation, and Modulation Theory*. New York, NY: John Wiley & Sons, 2002.

[89] Wang, L., Tao, J., Xiao, C., and Yang, T. C. "Frequency-Domain Turbo Detection for LDPC-coded Single-Carrier MIMO Underwater Acoustic Communications." *in Proceedings of MTS/IEEE Oceans conference (2010),* Seattle, WA, USA (2010): 1–5.

[90] Wang, S. and Abdi, A. "MIMO ISI Channel Estimation Using Uncorrelated Golay Complementary Sets of Polyphase Sequences." *IEEE Transactions Vehicular Technology* 56 (2007).5: 3024–3039.

[91] Wiener, N. *Extrapolation, Interpolation, and Smoothing of Stationary Time Series*. New York, NY: Wiley, 1949.

[92] Wolniansky, P. W., Foschini, G. J., Golden, G. D., and Valenzuela, R. A. "V-BLAST: An architecture for realizing very high data rates over the rich-scattering wireless channel." *in Proceedings of the ISSSE-98,* Pisa, Italy (September 1998): 295–300.

[93] Yang, S. and Wu, J. "Optimal Binary Training Sequence Design for Multiple-Antenna Systems Over Dispersive Fading Channels." *IEEE Transactions Vehicular Technology* 51 (2002).5: 1271–1276.

[94] Yang, T. C. "Temporal Resolutions of Time-Reversal and Passive-Phase Conjugation for Underwater Acoustic Communications." *IEEE Journal of Oceanic Engineering* 28 (2003).2: 229–245.

[95] ———. "Differences between passive-phase conjugation and decision feedback equalizer for underwater acoustic communications." *IEEE Journal of Oceanic Engineering* 29 (2004).2: 472–487.

[96] ———. "Channel Q function and capacity." *IEEE/MTS Oceans Conference* 1 (2005): 273–277.

[97] ———. "Correlation-Based Decision-Feedback Equalizer for Underwater Acoustic Communications." *IEEE Journal of Oceanic Engineering* 30 (2005).4: 865–880.

[98] ———. "Measurements of temporal coherence of sound transmissions through shallow water." *Journal of the Acoustical Society of America* 120 (2006): 2595–2614.

[99] Yang, T. C. and Al-Kurd, A. "Performance limitations of joint adaptive channel equalizer and phase locking loop in random oceans: Initial test with data." *IEEE/MTS Oceans Conference* 2 (2000): 803–808.

[100] Yardibi, T., Li, J., Stoica, P., Xue, M., and Baggeroer, A. B. "Source Localization and Sensing: A nonparametric iterative adaptive approach based on weighted least squares." *IEEE Transactions on Aerospace and Electronic Systems* 46 (2010): 425–443.

[101] Zangwill, W. I. *Nonlinear Programming: A Unified Approach*. Prentice-Hall, Inc., Englewood Cliffs, NJ 07632, 1969.

BIOGRAPHICAL SKETCH

Jun Ling received Bachelor of Science and Master of Science degrees in electrical engineering from Zhejiang University, Hangzhou, China, in 2004 and 2006, respectively. He graduated with a Doctor of Philosophy from the Electrical and Computer Engineering Department at the University of Florida in the summer of 2011. His research interests include signal processing and its application to multi-input multi-output underwater acoustic communications. He will join the Mathworks upon graduation.